

UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



TRABAJO FIN DE MÁSTER

Desarrollo de sistemas de sensores para teleasistencia

**Máster Universitario en Ingeniería de
Telecomunicación**

Autor: Víctor Manuel García López

Tutor: Alberto Sánchez González

Ponente: Ángel de Castro Martín

FECHA: FEBRERO 2016

Agradecimientos

Me gustaría comenzar este Trabajo Fin de Máster agradeciendo a todo aquel que ha ayudado a hacerlo posible. Empezando por todos los profesores que he tenido a lo largo del Grado y del Máster, durante los últimos 6 años, ya que sin sus enseñanzas no podría haber llegado hasta aquí.

Tengo que agradecerse especialmente a Ángel de Castro, que ha sido ponente tanto de mi Trabajo Fin de Grado como de mi Trabajo Fin de Máster, y a Alberto Sánchez, tutor de ambos proyectos también. También a Santiago Elvira, por darme la oportunidad de participar en este proyecto. Gracias a los tres por resolver siempre mis dudas.

Por último, me gustaría agradecerse a todos los compañeros, que han estado todos los días haciendo el camino más ameno, y también a mis amigos y especialmente a mi familia, por estar siempre ahí.

Resumen

Con el paso de los años, y el envejecimiento de la población, se ha llegado a una situación en la cual hay una gran cantidad de personas dependientes, pero el número de cuidadores disponibles no es tan alto. Esto ha llevado a la búsqueda de alternativas a la hora de cuidar a esta parte de la población.

Una de estas alternativas viene dada por el uso de las redes de sensores, que van a permitir a un solo cuidador monitorizar a diferentes personas al mismo tiempo de un modo mucho más cómodo, ya que ni siquiera es necesario que se encuentre en el hogar de dicha persona.

En el presente Trabajo Fin de Máster se pretende diseñar y fabricar un primer prototipo a partir del cual pueda crearse una red de sensores inalámbrica, cuyo principal objetivo será el de monitorización de personas dependientes. Mediante el uso de dicha red, podrán obtenerse datos sobre lo que está ocurriendo en el hogar monitorizado (temperatura de las distintas habitaciones, humedad, ventanas abiertas...) así como detectar situaciones anómalas (el anciano se levanta de la cama durante la noche y no vuelve en un tiempo razonable). Todos estos datos son enviados a un nodo central, que es el encargado de tomar las distintas decisiones sobre el sistema y enviarlos al exterior (enviar notificaciones al cuidador si alguna medida está fuera de lo normal, pedir medidas nuevas a los sensores, etc.).

Debido al uso al que pretende dedicarse el producto, habrá que tener en cuenta ciertos aspectos de cara al diseño del sistema, fundamentalmente su tamaño y sus consumos.

En este Trabajo Fin de Máster se mostrará cómo se ha llevado a cabo el diseño tanto hardware como software del primer prototipo, así como las diferentes decisiones sobre este que se han ido tomando. También se mostrarán las pruebas que se han realizado sobre el sistema para comprobar su correcto funcionamiento.

Palabras clave: red de sensores inalámbrica, monitorización de ancianos, comunicaciones inalámbricas, diseño PCB, interfaz gráfica, electrónica, Java, Arduino, I2C, circuito impreso.

Abstract

Over the years, and the aging of population, a situation has been reached where there is a great amount of dependant people whilst the number of available caregivers is not so big. This situation has led to seek alternatives to take care properly of this fraction of the population.

One of these alternatives comes as the use of sensor networks, which lets just one caregiver monitor different people at the same time on a more comfortable way, since he would not even have to be at the person's home.

The main goal of this work is designing and making a first prototype from which a wireless sensor network can be developed, and that will be aimed to monitor dependant people. By using such network, data about what is happening at the monitored home will be obtained (temperature on the different rooms of the house, humidity, open windows...). In addition, some anomalous situations could be detected (the elder gets up during the night and does not go back to bed in a reasonable time). All this data will be sent to a central node that will take the different decisions about the system (it will send notifications to the caregiver if some of the measures are out of the ordinary range, it will order the sensors to take new measures, etc.).

Because of the use the product will have, some aspects of the design will have to be taken into consideration, such as the size of the nodes or the consumption they will have.

In this work, the hardware and software designs, as well as the different decisions taken for the development of the first prototype of the system will be shown. In addition, the tests conducted to check the proper operation of the system will be shown.

Keywords: wireless sensor network, Elder monitoring, wireless communications, PCB design, graphic interface, electronics, Java, Arduino, I2C, printed circuit.

Índice

AGRADECIMIENTOS	I
RESUMEN	III
ABSTRACT	V
ÍNDICE	VII
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABLAS	XI
1. INTRODUCCIÓN	1
1.1. OBJETIVOS Y MOTIVACIÓN	2
1.2. ESTRUCTURA DE LA MEMORIA	5
2. ESTADO DEL ARTE	7
2.1. INTRODUCCIÓN	7
2.2. PROBLEMAS DE LAS REDES DE SENSORES	8
2.3. SOLUCIONES A LOS PROBLEMAS PROPUESTOS	9
2.4. EJEMPLOS DE REDES INALÁMBRICAS DE SENSORES	10
2.5. CASO ESPECÍFICO: MONITORIZACIÓN DE ANCIANOS	13
3. DISEÑO GENERAL	15
3.1. VISIÓN GLOBAL	15
3.2. NODO INALÁMBRICO	15
3.2.1. CAPA DE ALIMENTACIÓN	16
3.2.2. CAPA DE COMUNICACIÓN Y CONTROL	16
3.2.3. CAPA DE ADAPTACIÓN	16
4. ELECCIÓN DE COMPONENTES Y DISEÑO DE TARJETAS	17
4.1. NODO DE COMUNICACIONES Y CONTROL	17
4.2. DISEÑO DE TARJETAS	19
4.2.1. TARJETA DE ALIMENTACIÓN	19
4.2.2. TARJETA DE ADAPTACIÓN	20
4.2.3. TARJETA DE SENSORES	25
4.2.4. OTROS COMPONENTES	29

4.3. DISEÑO DE ESQUEMÁTICOS	31
4.3.1. TARJETA DE ALIMENTACIÓN ESTÁNDAR	31
4.3.2. TARJETA DE ALIMENTACIÓN DOBLE	31
4.3.3. TARJETA DE ADAPTACIÓN	32
4.3.4. TARJETA DE SENSORES	33
4.4. DISEÑO DE LAYOUTS	36
5. DESARROLLO SOFTWARE	37
5.1. CONTROL DEL MICROCONTROLADOR	37
5.1.1. COMUNICACIÓN CON LOS SENSORES	37
5.2. SOFTWARE PARA INTERFAZ Y CONTROL DEL SISTEMA	42
5.2.1. CAPA DE COMUNICACIÓN SERIE	43
5.2.2. CAPA DE COMUNICACIÓN INALÁMBRICA	43
5.2.3. CAPA DE GESTIÓN DE TARJETAS	45
5.2.4. CAPA DE ADAPTACIÓN	45
5.2.5. CAPA DE USUARIO	45
6. PRUEBAS Y RESULTADOS	47
6.1. COMUNICACIONES DENTRO DEL NODO INALÁMBRICO	47
6.1.1. SENSOR DE TEMPERATURA Y HUMEDAD	47
6.1.2. SENSOR TÉRMICO OMRON D6T	48
6.1.3. PCA9557	49
6.2. COMUNICACIONES INALÁMBRICAS	49
6.3. SISTEMA COMPLETO	50
7. CONCLUSIONES	53
8. TRABAJO FUTURO	55
ANEXO I: CERTIFICACIONES	57
ANEXO II: GLOSARIO	59
X. REFERENCIAS	61

Índice de figuras

Figura 1. Relación de niños y personas mayores dependientes ¹	1
Figura 2. Diagrama de bloques de la red a desarrollar	3
Figura 3. Diagrama de flujo de desarrollo del proyecto	4
Figura 4. Red de sensores inalámbrica (University of Surrey)	7
Figura 5. Diagrama de bloques de un nodo	8
Figura 6. Diseño global del sistema	15
Figura 7. Diseño del nodo inalámbrico	16
Figura 8. WiSMote (www.aragosystems.com)	17
Figura 9. WiSense (www.wisense.in)	17
Figura 10. MTM-CM3000-MPS (www.advanticsys.com)	18
Figura 11. PanStamp NRG 2 (www.panstamp.com)	18
Figura 12. Utilización de la tarjeta de alimentación doble	19
Figura 13. Soporte para pila	20
Figura 14. Tipos de conector USB	21
Figura 15. Conector micro-B	22
Figura 16. Sistema de elección de alimentación	23
Figura 17. Divisor resistivo	23
Figura 18. Esquema de lectura del CCD	27
Figura 19. Sensor Omron D6T	27
Figura 20. Esquemático de la tarjeta de alimentación estándar	31
Figura 21. Esquemático de la tarjeta de alimentación doble	31
Figura 22. Conectores, antena, divisor resistivo y resistencias pull-up de la tarjeta de adaptación	32
Figura 23. Esquemático del sistema de elección de alimentación	32
Figura 24. Esquemático de la conexión USB de la tarjeta de adaptación	33
Figura 25. Esquemático de los reguladores de tensión	33
Figura 26. Conector y test points de la tarjeta de sensores	33
Figura 27. Esquemático de los interruptores de carga de la tarjeta de sensores	34
Figura 28. Esquemático del conector para el sensor Omron D6T junto al convertidor de nivel I2C	34
Figura 29. Esquemático del sistema de identificación de tarjetas	34
Figura 30. Esquemático de los sensores de humedad y temperatura	35
Figura 31. Esquema de comunicación del HIH6030	38
Figura 32. Esquema de comunicaciones del sensor Omron D6T	39
Figura 33. PCA9557: Introducir configuración de E/S	40
Figura 34. PCA9557: Lectura de registro	40
Figura 35. Configuración de polaridad	40
Figura 36. Diagrama UML del software desarrollado	42
Figura 37. Nodo fabricado	47
Figura 38. Medidas de temperatura y humedad frente al tiempo	48
Figura 39. Prueba del sistema completo: temperatura	50
Figura 40. Prueba del sistema completo: humedad	51
Figura 41. Prueba del sistema completo: Omron (primera medida)	51
Figura 42. Prueba del sistema completo: Omron (segunda medida)	51

Índice de tablas

<i>Tabla 1. Modelos de nodos de redes de sensores inalámbricas</i>	12
<i>Tabla 2. Modelos USB-UART considerados</i>	22
<i>Tabla 3. Reguladores de tensión considerados</i>	24
<i>Tabla 4. Modelos de los expansores de entrada/salida considerados</i>	26
<i>Tabla 5. Modelos de interruptores de carga considerados</i>	28
<i>Tabla 6. Modelos de sensores de humedad y temperatura considerados</i>	29
<i>Tabla 7. Modelos de ADC considerados</i>	29
<i>Tabla 8. Tipos de paquete para comunicaciones inalámbricas</i>	44
<i>Tabla 9. Prueba sobre el sensor Omron D6T</i>	49

1. Introducción

En las últimas décadas, ha aumentado enormemente la cantidad de población de avanzada edad debido a los avances médicos y el aumento en la calidad de vida. Con el aumento de este tipo de población, nos encontramos ante una situación en la que hay una gran proporción de personas mayores a las que cuidar, de modo que los recursos disponibles pueden no ser suficientes.

Además, esta situación no va a cambiar en un futuro cercano¹. Gracias a la disminución de la mortalidad acompañada de una reducción en la fertilidad, la proporción de personas mayores en la población está creciendo. Es decir, la población está envejeciendo.

Como se muestra en¹, en las últimas seis décadas la esperanza de vida ha aumentado en unos 18 años (de 47 años en 1950-1955 a 65 años en 2000-2005). Previsiblemente llegará hasta los 75 años en torno al 2050. Por el contrario, la fertilidad ha ido disminuyendo de 5,0 hasta 2,6 hijos por mujer, y se prevé que continúe disminuyendo hasta los 2,0 hijos por mujer en 2050.

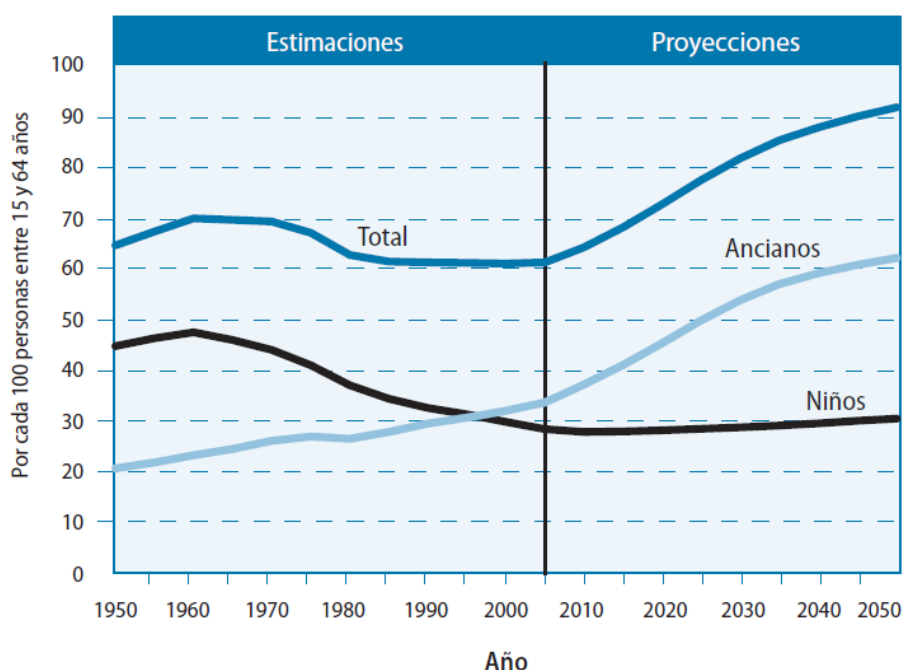


Figura 1. Relación de niños y personas mayores dependientes¹

En la Figura 1 podemos observar la proporción de niños (menores de 14 años) y personas mayores (mayores de 64 años), en relación con la población entre 15 y 64 años. La suma de ambas se muestra como la relación de dependencia total. Observando las variaciones en las relaciones de dependencia, se puede comprobar como en 2005 aproximadamente se produjo un mínimo en el porcentaje de personas dependientes (incluyendo niños y ancianos) y a partir de ese punto ha ido aumentando, esencialmente, en la cantidad de personas mayores dependientes.

Esto supone una disminución en la cantidad de fuerza laboral que se puede encontrar y, por tanto, se traduce en una reducción de la cantidad de cuidadores que podrán encontrarse. Para tratar de facilitar el cuidado de las personas mayores, así como para intentar solventar los

problemas ya expuestos, cada vez son más las investigaciones que se llevan a cabo en el campo de las nuevas tecnologías para ser aplicadas a este campo.

Para ello, se han desarrollado gran cantidad de sistemas que tratan de controlar las condiciones del entorno de los hogares de estas personas^{2, 3}, monitorizar sus constantes vitales^{4, 5}, detectar caídas^{5, 6}, etc. Incluso con algunos sistemas se ha tratado de proporcionar a los ancianos alguna forma de comunicarse, de modo que no se sientan solos, mediante el uso de robots^{6, 7} o la inclusión de salas virtuales en sus hogares⁸.

Esta clase de métodos utilizados suelen ser invasivos (suelen utilizar cámaras para vigilancia) o muy costosos. Otros son más difíciles de usar o suponen la instalación de dispositivos de un tamaño considerable en los hogares de estas personas.

El sistema a desarrollar en este Trabajo Fin de Máster está basado en redes de sensores inalámbricas, y busca mitigar los inconvenientes del resto de sistemas que se han desarrollado hasta la fecha. Para ello, en lugar de utilizar cámaras de videovigilancia, se utilizarán solamente sensores que no supongan una invasión de la privacidad de los ancianos, a la vez que garantizan su seguridad. Se tendrán en cuenta también otros aspectos como el tamaño, alimentación eléctrica, consumos, etc.

Por último, puesto que el dispositivo a crear tiene un objetivo comercial, será necesario tener en cuenta los distintos estándares necesarios para poder comercializar dicho producto en los mercados objetivo, que serán Europa y Estados Unidos, y además se intentará reducir el precio tanto como sea posible, de forma que el producto no resulte demasiado costoso.

1.1. Objetivos y motivación

Los sistemas de monitorización de ancianos suponen un método muy eficiente para facilitar el cuidado de éstos, así como para dar la opción de que un solo cuidador pueda encargarse de varios de ellos. Desarrollando un sistema que permita a los ancianos sentirse seguros en sus casas al mismo tiempo que se mantiene íntegra su intimidad, se podrá mejorar la calidad de vida de estas personas.

Siguiendo estas motivaciones, se intentará crear una red de sensores inalámbrica que permita llevar a cabo la monitorización de una casa tomando diferentes medidas, que dependerán del caso específico a analizar. Mediante este sistema se conseguiría minimizar el tiempo invertido por los cuidadores para realizar el seguimiento del paciente, siendo posible llevar a cabo este a través de los dispositivos instalados, y detectando cualquier situación anómala que pudiese ocurrir.

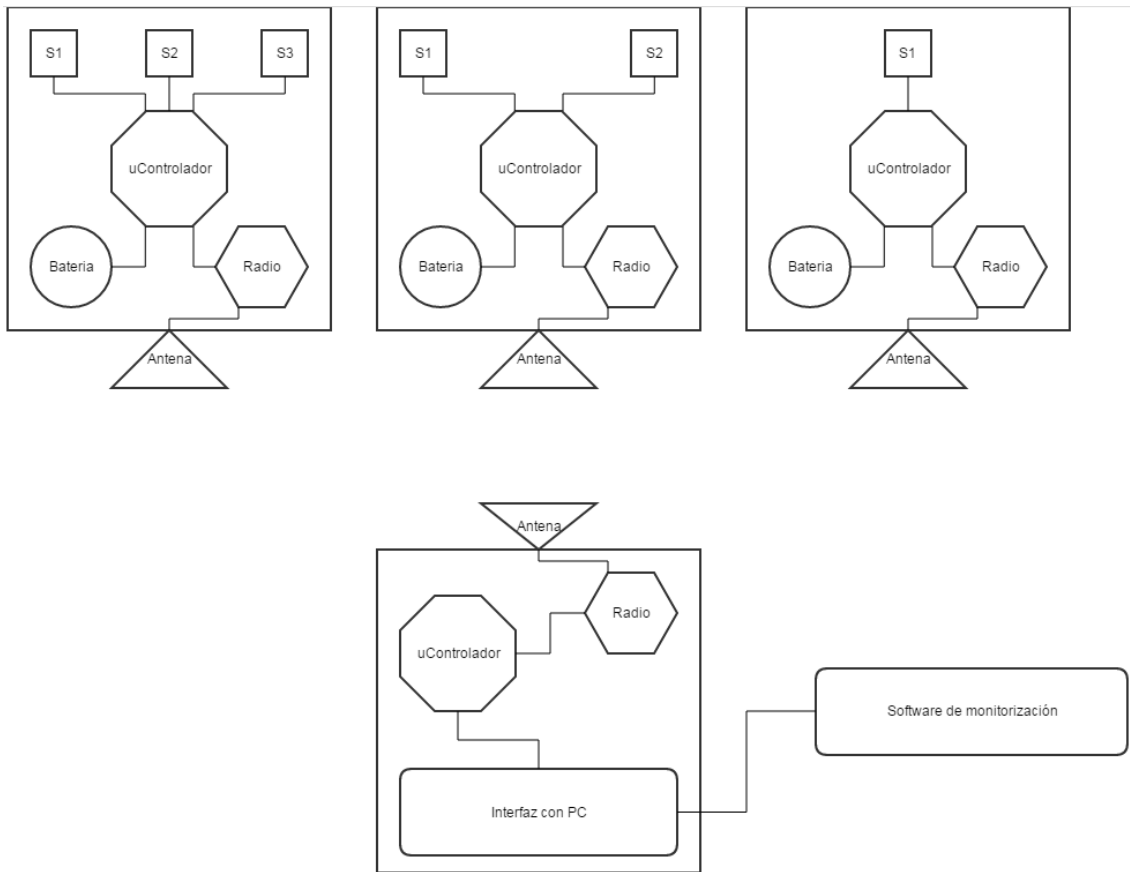


Figura 2. Diagrama de bloques de la red a desarrollar

Para ello, se creará una plataforma modular similar a la mostrada en la Figura 2, que permita utilizar diferentes sensores, nombrados como S_x , donde x es un número, cuyos datos serían procesados por el microcontrolador, y después enviados al nodo central, que será el encargado de llevar a cabo la comunicación con el cuidador. Los diferentes nodos se alimentarán generalmente mediante baterías. La principal novedad del sistema con respecto a los ya existentes es el uso de un sensor de presencia térmico. Se trata de un CCD de muy baja resolución que captura la temperatura en una determinada área. De este modo, se pueden detectar gran cantidad de situaciones anómalas (si una persona se levanta de la cama en mitad de la noche y no vuelve, o si no se ha levantado a una hora determinada, si ha salido a la calle y no ha vuelto en un cierto tiempo...), sin vulnerar en ningún caso la intimidad de la persona gracias al tipo de sensor y a su baja resolución, ya que al contar con solo 16 píxeles no es posible obtener información sobre qué actividad está realizando la persona.

El objetivo principal de este Trabajo Fin de Máster es diseñar el nodo inalámbrico a utilizar para este tipo de aplicación. Para ello, se seguirá el diagrama de flujo que puede observarse en la Figura 3.

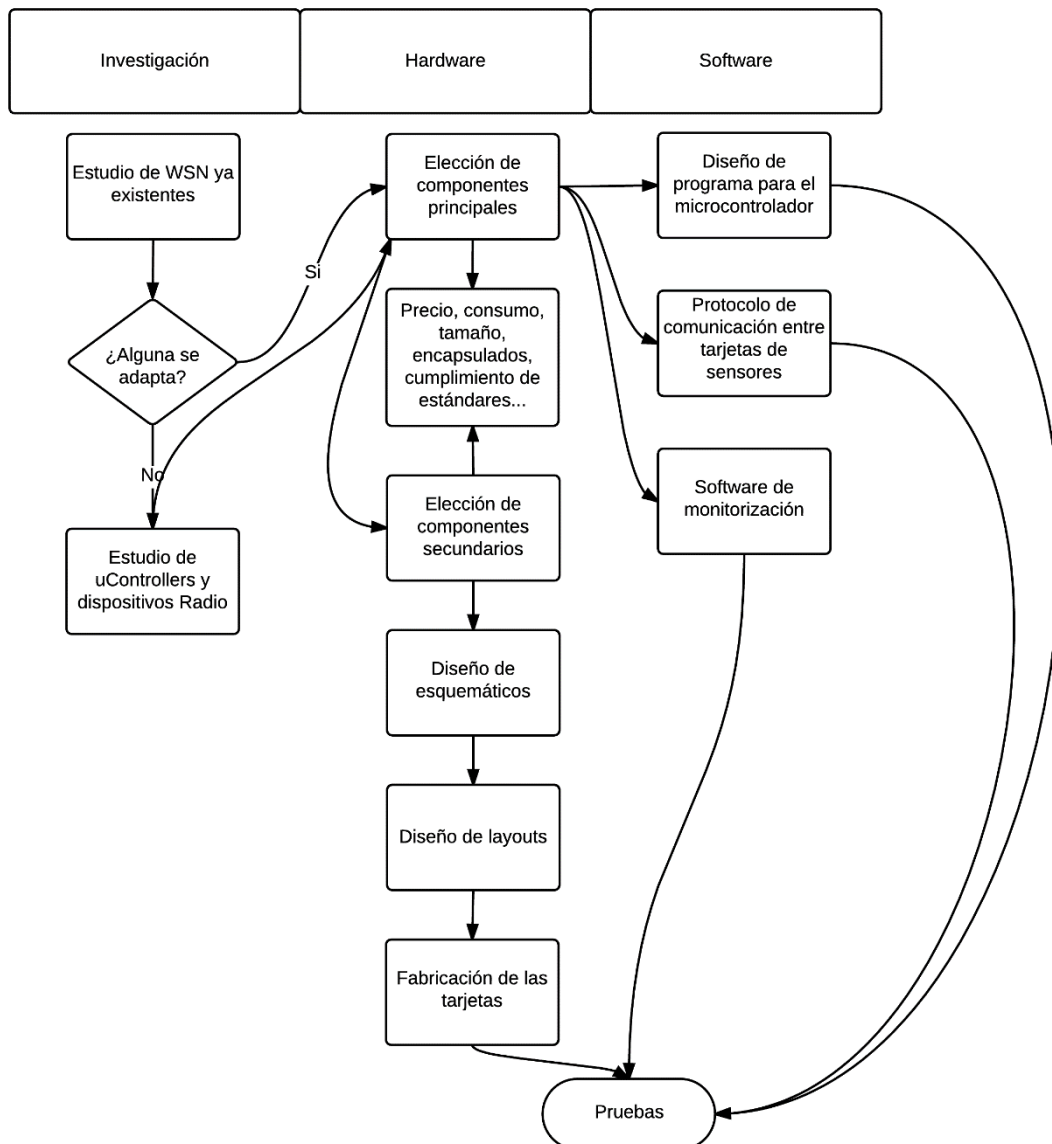


Figura 3. Diagrama de flujo de desarrollo del proyecto

Habrá que, en primer lugar, investigar los productos ya existentes en el mercado de forma que pueda decidirse qué opción es más recomendable: diseñar el nodo completo desde el principio, o bien utilizar alguno de los ya creados y añadir las modificaciones oportunas para que se adapte a las necesidades requeridas.

También deberán elegirse los sensores a utilizar para la aplicación, así como llevar a cabo su adaptación dentro del nodo inalámbrico, de forma que pueda utilizarse un zócalo estándar, para así poder desplegar la red de sensores utilizando una tarjeta base común en lugar de tener que construir una distinta para cada variante del sistema a desarrollar.

En principio, deciden colocarse sensores de temperatura y humedad, así como el sensor CCD. Dependiendo del tamaño de los sensores, se decidirá más tarde si es necesario crear una tarjeta para cada uno de los sensores o si se colocarán en la misma.

Además, puesto que se quiere lograr que el dispositivo creado sea de uso sencillo, se incluirá en cada tarjeta de sensores un código que identifique qué tipo de tarjeta es, y que a la vez permita usar más de una tarjeta del mismo tipo al mismo tiempo.

También debe implementarse un sistema en las tarjetas que permita llevar diversas tensiones de alimentación, cuyos valores dependerán de los componentes que se elijan finalmente para el diseño.

Por último, el objetivo de este trabajo es conseguir la creación de un producto comercial que pueda venderse en Europa y Estados Unidos más tarde, por tanto es importante que el dispositivo cumpla con los estándares necesarios para su comercialización en estos lugares (RohS, FCC, CE...).

Nota: Pese a que este primer prototipo no se homologará, las restricciones definidas por estos estándares serán un aspecto a tener en cuenta de cara al diseño y la elección de componentes del sistema.

1.2. Estructura de la memoria

La estructura de esta memoria se ha dispuesto como se explica a continuación:

- En el capítulo 1, se explica la motivación y los objetivos que se pretenden lograr en el desarrollo de este proyecto, y se muestra la estructura que va a seguir toda la memoria, así como cómo se llevará a cabo el desarrollo del proyecto.
- En el capítulo 2 se mostrará un resumen del estado del arte sobre redes inalámbricas de sensores, así como de las aplicaciones posibles en las que se pueden encontrar, y se entrará en detalle sobre los distintos modelos de nodos existentes en el mercado y las investigaciones llevadas a cabo sobre la aplicación a la que se orienta este proyecto.
- En el tercer capítulo, se describirá el diseño general del dispositivo. Fundamentalmente se darán detalles sobre cómo se realizará la disposición de las diferentes tarjetas que quieran fabricarse y añadirse al diseño.
- Se detallará a continuación la elección de componentes específicos, así como el diseño de las propias tarjetas (incluyendo esquemáticos y *layouts*).
- En el capítulo 5 se mostrará cómo se ha realizado el proceso de desarrollo de los programas software necesarios para llevar a cabo el control del dispositivo, así como su comunicación con el resto de dispositivos y con el PC desde el que se pretende monitorizar toda la red.
- En el capítulo 6 se mostrarán las pruebas realizadas de cara a verificar que el diseño funciona como se esperaba que lo hiciese.
- En el séptimo capítulo se mostrarán las conclusiones a las que se ha llegado con la realización del proyecto
- Para finalizar, en el capítulo 8 se mostrarán algunas líneas posibles de investigación futura.

2. Estado del arte

Cada día es más natural encontrar diversos dispositivos conectados mediante medios inalámbricos, desde los más comunes, como ordenadores o *smartphones*, hasta otros que no lo son tanto, como tostadoras o cafeteras.

Con el desarrollo de las tecnologías *wireless*, no solo se ha permitido conectar estos objetos cotidianos a la red, sino que cada vez se encuentran más usos para ellas. En el caso de este Trabajo de Fin de Máster (TFM en adelante), son las redes de sensores inalámbricas las que nos incumben. Un ejemplo de este tipo de redes puede observarse en la Figura 4.

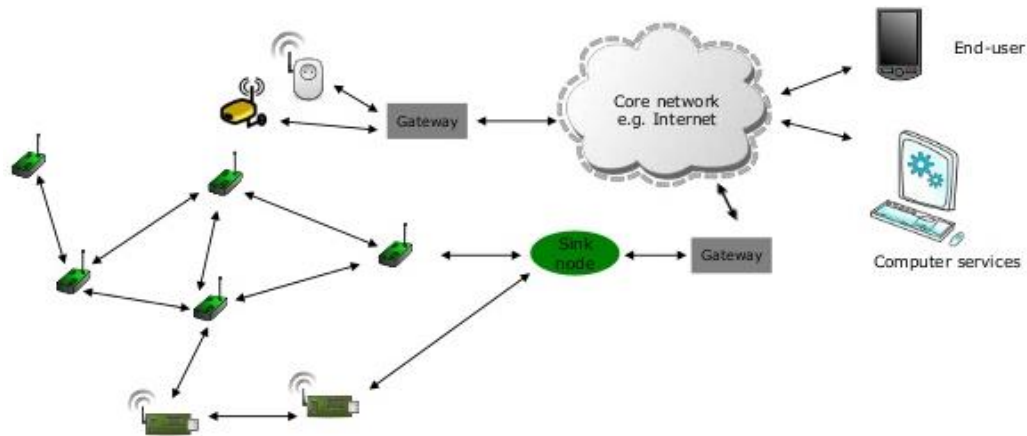


Figura 4. Red de sensores inalámbrica (University of Surrey)

2.1. Introducción

Las redes de sensores inalámbricas o WSN (del inglés, *Wireless Sensor Networks*) son un conjunto de dispositivos que van a permitir realizar medidas del entorno en una gran variedad de situaciones.

Están formadas generalmente por una fuente de alimentación, un microcontrolador, que va a ser el encargado de controlar las comunicaciones que sean necesarias entre los distintos dispositivos, un transceptor, que permitirá realizar la comunicación inalámbrica del dispositivo y un cierto número de sensores, que variarán en tipo (humedad, temperatura, gas...) y cantidad (redes para casas, para empresas, para control medioambiental...) dependiendo de la aplicación específica a la que estén destinadas. En la Figura 5 puede observarse un diagrama de bloques de un nodo típico.

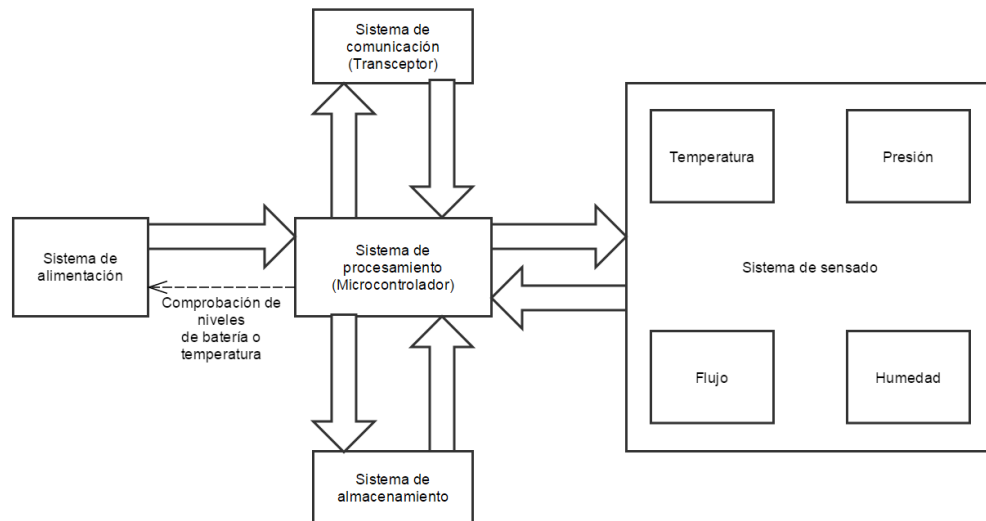


Figura 5. Diagrama de bloques de un nodo

Debido a sus principales aplicaciones, suele ser necesario que estos dispositivos sean autónomos, por tanto debe minimizarse su consumo para que puedan alimentarse con baterías. Además, de nuevo debido a sus aplicaciones, suelen necesitarse diseños de bajo coste y tamaño reducido.

Los usos más destacados de esta tecnología son:

- Monitorización del medioambiente
- Monitorización de condición estructural (daños por fatiga en puentes de carretera...)
- Monitorización de condición de máquinas
- Monitorización de nivel de ocupación de un *parking*
- Monitorización de pacientes en un centro médico
- Monitorización de las condiciones en un determinado lugar de trabajo (temperatura, humedad de un taller o fábrica...)
- Hogares inteligentes

2.2. Problemas de las redes de sensores

Como ya se ha adelantado, uno de los principales problemas que se presentan de cara a la utilización de estas tecnologías es la necesidad de hacer funcionar el dispositivo mediante baterías. Si bien algunos componentes como los sensores no consumen demasiada energía, el microcontrolador y el adaptador de radiofrecuencia sí que lo hacen.

El principal problema es, que si bien el desarrollo de las nuevas tecnologías, así como la capacidad de procesamiento y comunicación de dispositivos se ha venido produciendo a grandes pasos las últimas décadas, no ha ocurrido lo mismo con las baterías. De hecho, la alimentación que utilizan los últimos modelos de WSN es, en la mayoría de los casos, la misma que utilizaban los modelos de hace casi dos décadas.

Esto además, trae consigo otro problema. Si la batería de los dispositivos se agota constantemente se hace necesario reemplazarla de forma habitual, lo que hace que aumente el costo de esta tecnología enormemente. Aun si se consiguiese recargar las baterías de forma

automática, seguiría siendo un problema (aunque de menor grado) ya que las baterías actuales tienen un tiempo de vida limitado, y sería necesario reemplazarlas cada cierto tiempo.

Otro importante problema de cara a esta tecnología es el tamaño de los dispositivos. Para ciertas aplicaciones, como puede ser el control medioambiental, éste no es un factor demasiado importante. Sin embargo, para otras como la monitorización de pacientes, es necesario crear dispositivos de tamaño muy reducido, y además con mucha movilidad, de forma que los pacientes puedan llevarlos encima sin resultar demasiado molestos.

Del mismo modo, para aplicaciones diseñadas para hogares, es conveniente que el tamaño del dispositivo sea reducido, ya que de este modo se mejora la aceptación por parte de los clientes del producto.

2.3. Soluciones a los problemas propuestos

Muchas son las investigaciones que se han realizado de cara a solucionar los problemas relacionados con las WSN que, si bien han conseguido demostrar que hay forma de enfrentarse a ellos, no han logrado dar una solución del todo válida.

En general, y cuando la aplicación lo permite, se suele optar por el uso de paneles solares para proporcionar energía extra al dispositivo. Para este tipo de dispositivos, se han llevado a cabo investigaciones⁹ sobre cómo almacenar la energía proporcionada por estos paneles, de cara a poder ser utilizada más tarde, de modo que teóricamente podría alargarse el tiempo de operación de un determinado sistema hasta 20 años. Por supuesto, para que esto sea posible, el consumo del dispositivo debe de ser muy reducido. Del orden de los milivatios.

También se han llevado a cabo investigaciones^{10, 11} sobre otras formas de alimentar estos dispositivos de forma inalámbrica, desarrollando sistemas que recojan potencia de la energía microondas, pudiendo así alimentarse a distancia, ya sea recargando las baterías del dispositivo cuando sea necesario, o directamente alimentando los componentes sin necesidad de baterías.

Por otra parte, y de cara a reducir el consumo o emplear más eficientemente los recursos, se han realizado también diversos estudios. Una interesante funcionalidad que añadir a estos dispositivos, sería la de medir la cantidad de energía almacenada en ellos, de forma que, cuando sea baja, pueda intentarse disminuir el consumo del sistema, de modo que no deje de funcionar en ningún momento. En la referencia¹² puede encontrarse más información al respecto sobre cómo realizar esta tarea sobre las baterías más utilizadas actualmente para alimentar estos sistemas.

También se han investigado diversas formas de disminuir el consumo de los sistemas. Uno de los modos más interesantes de lograrlo¹³, es mediante el diseño de una arquitectura de procesador específica, que puede consumir de 8 a 10 veces menos energía que uno de los procesadores que se utiliza típicamente, y realizar las tareas de 2 a 3 veces más rápido, lo que se traduce en un ahorro de energía aún mayor.

Muchos otros modos de ahorrar energía han sido estudiados, desde un mejor aprovechamiento de las baterías mediante el uso de un condensador en paralelo con ellas¹⁴, pasando por incluir un piezoeléctrico que permita generar energía eléctrica que, aunque pequeña, sea suficiente para aumentar la vida útil del dispositivo¹⁵, hasta la compartición de energía entre sensores¹⁶.

2.4. Ejemplos de redes inalámbricas de sensores

Dentro del campo de los nodos para las redes inalámbricas de sensores, hay un amplio número de dispositivos que se han ido creando con el paso de los años. Generalmente, estos dispositivos se han desarrollado sin pensar en una aplicación específica, permitiendo conectar gran variedad de sensores a cada uno de ellos, y permitiendo elegir parte de las especificaciones del producto (banda de trabajo del adaptador de radiofrecuencia programable, posibilidad de elección de la frecuencia de trabajo...).

A continuación se detalla un estudio que se ha realizado sobre la evolución de dichos modelos con el paso de los años, junto a una tabla (Tabla 1) que resume dicho estudio. Hay que tener en cuenta que la mayoría de estos nodos se han desarrollado a nivel experimental, y solo algunos de ellos se encuentran a nivel comercial:

- **Rene1¹⁷**: Se trata de un primer acercamiento a esta tecnología. Se realizó una hipótesis sobre los requerimientos basándose en aplicaciones potenciales, y eligiendo el tamaño en consecuencia. Se trató de realizar un diseño pequeño, que consumiese poca potencia, sin cables, y que fuera autoconfigurable.
- **Awairs1¹⁸**: Se trata de un dispositivo con diseño modular y abierto basado en COTS (es decir, productos *off-the-shelf*, o productos comerciales disponibles para ensamblar). Se intentaba realizar procesamiento de señal embebido, de forma que se pudiesen disminuir las necesidades de comunicación, y así ahorrar energía. Se propuso el uso de Java para el microcontrolador, pero se descartó debido al consumo que produciría. Era un modelo poco eficiente en tamaño, consumo y coste, pero que presentaba todas las características esenciales, y que representaba el vehículo ideal para las investigaciones futuras.
- **Mica¹⁹**: Se diseña un sistema con protocolos muy eficientes en tiempo y consumo. Es una plataforma experimental que combina comunicación, computación, control de consumo y sensado. Se consigue una corriente de *standby* de unos pocos microamperios. Contaba además con un bus de extensión que permitía un gran número de conexiones (SPI, I2C...).
- **Mica2²⁰**: Similar al modelo anterior, pero incluye un nuevo modelo de transceptor, que va a permitir el uso de diferentes canales de frecuencia, además de un sistema operativo más actualizado, y permite realizar reprogramación remota inalámbrica. También incluye un modelo distinto de microcontrolador.
- **Mantis Nymph²¹**: Se crea a partir de Mica. Contiene todas las características que serían de esperar en una WSN. Soporta un gran número de interfaces distintas, y añade una interfaz JTAG para poder cargar controladores en el hardware. El sistema operativo que utiliza es MANTIS OS, que permite a la WSN tres modos de funcionamiento distinto, que van a producir ahorros en el consumo de energía del dispositivo. Este modelo además incluía la posibilidad de añadir sensores GPS, para poder obtener, además de las medidas de los sensores, la posición exacta de cada una de estas. Se intentó dar soporte a usuarios con diversos niveles de experiencia en programación.
- **ProSpeckz²²**: Se trata de un dispositivo con un tamaño muy reducido. Para la alimentación, a diferencia de los modelos anteriores que solían confiar en baterías típicas AA, se utiliza una pila de botón de 3 V. Se consiguió crear un dispositivo que permitía realizar test físicos sobre radio MAC y protocolos a nivel de red. Permitía dar estimaciones de funcionamiento, requerimientos y restricciones, y era fácilmente extensible mediante conexión de hardware adicional. Aun así, sus recursos eran muy

limitados, y se trataba de un entorno poco familiar para la programación, lo que no era llamativo para usuarios potenciales.

- **Pluto²³**: Este dispositivo estaba orientado a uso médico. Se trata de pequeños dispositivos móviles que puedes llevar encima y que controlan tus señales vitales, lo que permitiría a doctores, enfermeros o cuidadores en general monitorizar el estado de sus pacientes constantemente. Se trata de un sistema que tiene que ser por necesidad extremadamente pequeño, lo que hace que los dispositivos creados hasta la fecha distasen bastante de lo que se quería conseguir en este caso. Para conseguir los objetivos, se sacrificó la expansibilidad del dispositivo, así como la duración de la batería. Era posible utilizarlo de forma continua hasta 5 h, aunque los tiempos aumentaban considerablemente mediante el uso de los modos de bajo consumo.
- **DSYS25²⁴**: Se intentan desarrollar unidades sensoras autónomas en miniatura, que puedan utilizarse y mantenerse en el entorno. Se desarrolló una versión del sistema operativo TinyOS específica, que fuese capaz de controlar cuidadosamente el consumo de potencia. Se utilizó un transceptor que era capaz de realizar CRC y computación de direcciones, de forma que se evitaba despertar (salir del modo de *standby*) al microcontrolador para estas tareas, permitiendo así un mayor ahorro de energía. Además se consiguió tener velocidades de transmisión de datos de hasta 1 Mbps, lo que a su vez permitía enviar los datos más rápidamente, y por tanto, al disminuir el tiempo de transmisión, disminuir también el consumo de energía.
- **AquisGrain²⁵**: Se orientó principalmente a aportar un modo eficiente del control de energía. Se trata de una plataforma orientada a la monitorización de pacientes. Contaba con gran cantidad de interfaces para conexión, y con una línea encargada de monitorizar la batería del dispositivo.
- **ScatterWeb ESB²⁶**: Se desarrolló con fines de investigación. Incluía un gran número de sensores comunes, así como algunos actuadores que, aunque resultaban útiles para investigación y enseñanza, resultaron ser bastante pesados. Estaba formado por una placa central, a la que podían conectarse otras a los lados. La alimentación se suministraba por medio de otra placa externa. Permitía la elección del canal de radiofrecuencia a utilizar, así como de la potencia, lo que daba posibilidad al ahorro de energía.
- **Shimmer²⁷**: Se trata de un dispositivo formado por una placa base, que se encargaba de las actividades de computación, comunicación, almacenamiento y que podía conectarse a diferentes placas inferiores, permitiendo extender su funcionalidad.
- **Neomote²⁸**: Se consiguió un consumo de potencia extremadamente reducido, con un diseño poco ruidoso. Permitía programar el microcontrolador mediante JTAG, y contaba además con interfaces muy variadas.
- **Arduino BT²⁹**: Basado en tecnología Bluetooth. Puede ser programado inalámbricamente mediante el uso de dicha tecnología. De nuevo, permitía el uso de gran cantidad de interfaces, y tenía la principal diferencia de que, en este caso, la alimentación se le suministraba por cable.
- **Indriya_DP_03A14³⁰**: Cuenta con una placa principal, que contiene los componentes principales, y proporciona la posibilidad de conectar otras placas con sensores, permitiendo una alta adaptabilidad en la construcción de sistemas. A diferencia de la mayoría de dispositivos, utiliza para comunicación por radiofrecuencia un módulo WLAN, de bajo consumo y uso sencillo, y la comunicación estaba basada en TPC/IPv4.

La placa principal contaba con varios sensores de muy bajo consumo (temperatura, luz y vibración). Contaba con gran variedad de interfaces posibles.

- **FireFly³¹**: Se intentaba desarrollar un modo de sincronización en el tiempo, de forma que no solo se tuviese constancia de los datos recogidos, sino del orden en que estos ocurrían sin necesidad de despertar al procesador para ello. Para tal cometido, utiliza dos relojes, uno de 16 MHz que controla el transceptor y la CPU, y otro de 32 MHz, de bajo consumo, que funciona todo el tiempo (incluso con el procesador en *standby*). Si bien se consiguió el objetivo de la sincronización, con unos consumos mínimos en estado inactivo (22 μ W), el hardware adicional necesario empleado para la sincronización supone un aumento considerable tanto en coste, como en gasto de energía.

Dentro de los modelos más modernos, cabe destacar tres de ellos especialmente:

- **Wismote/Wismote mini³²**: Se trata de un dispositivo de tamaño y consumo muy reducidos. Se alimenta con una batería de botón y soporta gran cantidad de interfaces de comunicación. Además alcanza unas velocidades considerables en transmisión de datos.
- **WiSense³³**: Se trata de nuevo de un dispositivo de tamaño muy reducido, con un diseño modular. Está dividido en tres placas distintas, una primera, que contiene el microcontrolador, así como comparadores de voltaje, ADC, y gran cantidad de interfaces de comunicación, y cuenta además con un sensor de luz y otro de temperatura, y permite monitorizar la temperatura y el voltaje de alimentación. Por otra parte está el módulo de radio, formado por una primera placa base, que será la que realice la conexión entre la placa principal y la radio, y la placa radio, con el transceptor y la antena.
- **PanStamp NRG³⁴**: Se trata de uno de los modelos más pequeños que se pueden encontrar, contando con gran cantidad de interfaces para comunicación. Permite además elegir parámetros del dispositivo como la frecuencia de trabajo o la radiofrecuencia a la que realizar las comunicaciones.

Nombre	Empresa	Año	Microcontrolador	Transceptor
Rene1	Universidad de California, Berkeley	1999	Atmel AT90LS8535	RF Monolithics TR1000
Awairs1	Rockwell	1999	Intel StrongARM SA-1100	RDSSS9M
Mica	Universidad de California, Berkeley	2002	Atmega103L	RF Monolithics TR1000
OMica2	MEMSIC	2003	Atmel ATmega128L	ChipCon CC1000
Mantis Nymph	Universidad de Colorado	2003	Atmel ATmega128L	ChipCon CC1000
ProSpeckz	Universidad de Edimburgo	2004	Atmel AT91	ChipCon CC2420
Pluto	Universidad de Harvard	2004	TI MSP430F149	ChipCon CC2420
DSYS25	University College Cork, Irlanda	2004	Atmel ATmega128L	Nordic nRF2401
AquisGrain	Philips Research	2004	Atmel ATmega128L	ChipCon CC2420
ScatterWeb ESB	Freie Universität Berlin	2005	TI MSP430F149	Chipcon CC1020
Shimmer	Intel	2006	TI MSP430F1611	ChipCon CC2420
Neomote	Crossbow/MEMSIC	2007	Atmel ATmega128L	Dust Networks Eterna

Arduino BT	Arduino	2008	Atmel ATmega328	2.1 WT11i-A-AI4
Indriya_DP_03A14	Indrion	2010	Atmel ATmega128L	Zigbee radio module, XBEE-Series2
FireFly3	Carnegie Mellon University	2012	Atmel ATmega128RFA1	Integrado en el chip del procesador
Wismote (mini)	Arago Systems	2012	Atmel ATmega128RFA2	Integrado en el chip del procesador
WiSense	WiSense Technologies	2014	TI MSP430	ChipCon CC1101
panStamp NRG	panStamp	2015	TI MSP430	ChipCon CC1101

Tabla 1. Modelos de nodos de redes de sensores inalámbricas

2.5. Caso específico: monitorización de ancianos

Puesto que el sistema que se va a desarrollar está pensado para monitorización de personas mayores que viven solas en sus casas, resulta interesante comprobar qué otros sistemas se han desarrollado para tales fines.

Para este tipo de situaciones, el método más empleado consiste en monitorizar la corriente que llega a los diferentes electrodomésticos que pueda haber en la casa del anciano. De esta forma, se colocarían sensores de corriente en varios de ellos y sería posible saber cuándo están siendo utilizados^{35, 36 y 37}. En caso de no poder conectar todos, que suele ser lo habitual, se tiende a utilizar la televisión, el lavavajillas y el frigorífico, que suelen ser los más utilizados por las personas de mayor edad.

A parte de este método, también se suele optar por colocar sensores de presión en la cama de la persona, de forma que se pueda saber si se encuentra en ella o no^{36 y 37}. Este método también se amplía en ocasiones a otro mobiliario, como pueden ser sillas, o incluso el inodoro.

Otro método habitual de monitorización consiste en controlar el flujo de agua en los grifos de la casa, de esta forma se puede tener constancia de cuando son utilizados^{36 y 37}. Otros métodos menos habituales sobre los que se ha investigado consiste en la detección de caídas³⁸, o la unión de sensores estáticos, como los descritos anteriormente, junto a sensores móviles que los propios ancianos llevarían encima³⁹, permitiendo tener constancia no solo de si la persona presenta un comportamiento normal, sino también permitiendo ver que todas sus constantes sean correctas. Incluso se han desarrollado sistemas capaces de identificar los periodos habituales de actividad en un hogar, y detectar cuando ocurre un periodo de inactividad poco común, permitiendo enviar una alerta a cuidadores o familiares⁴⁰.

En cualquier caso, el uso de la gran mayoría de estos métodos supone conocer los patrones de comportamiento de la persona o personas a monitorizar, lo que supone una dificultad bastante importante ya que, en primer lugar, se hace necesario tener un periodo de aprendizaje de patrones comunes por parte del sistema. Tras este periodo, puede comenzar su funcionamiento, pero es necesario desarrollar un algoritmo complejo (y por tanto costoso) de cara a analizar las diferentes situaciones que puedan ocurrir, y clasificarlas de forma correcta como una señal de que ocurre algún problema, o bien como simplemente una salida del patrón habitual, de forma que no se produzcan falsas alarmas innecesariamente.

Otros sistemas más sencillos y menos costosos optan por algunos de los métodos anteriormente descritos, como sensores de presión o de flujo, a la vez que incluyen sensores de temperatura, gas o incluso uso de puertas, que permiten comprobar que no hay problemas sin necesidad de prestar tanta atención a los patrones de comportamiento⁴¹.

3. Diseño general

El objetivo del presente TFM es desarrollar un sistema de sensores inalámbrico destinado a la monitorización de personas mayores. Debido a su campo de aplicación, habrá que tener diversos factores en cuenta de cara al diseño del sistema, especialmente el tamaño, el precio y el consumo, de forma que se consiga la mayor autonomía posible para el sistema.

Adicionalmente, se tratará de crear un diseño que permita trabajar de forma sencilla con el sistema completo, permitiendo al usuario personalizar el producto dependiendo de sus necesidades sin tener que conocer los detalles de la implementación.

3.1. Visión global

El esquema básico a seguir para tener el sistema en funcionamiento puede observarse en la Figura 6.

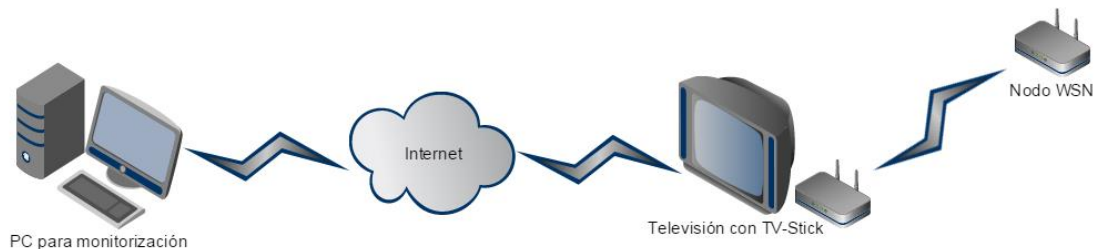


Figura 6. Diseño global del sistema

Fundamentalmente, se creará un nodo *wireless* que permita realizar una conexión con la televisión de la persona mayor (idealmente mediante el uso de un TV-Stick). El nodo inalámbrico incluirá una serie de sensores que enviarán sus medidas cada cierto tiempo. Estos datos se recibirán por parte del cuidador, que podrá controlar en todo momento que las medidas se ajustan a las esperadas.

En caso de producirse algún problema (temperatura o humedad por encima o por debajo de los valores esperados, el anciano lleva más tiempo del habitual tumbado en la cama o lleva muchas horas sin moverse del sofá...) el cuidador podrá tomar las medidas necesarias para asegurarse de que todo va bien.

3.2. Nodo inalámbrico

Para crear el nodo inalámbrico se pretende crear un diseño sencillo de utilizar para el usuario, de forma que éste pueda hacer funcionar el sistema sin necesidad de poseer conocimientos técnicos. Para ello, se decide utilizar un diseño modular dividido en cuatro capas distintas, cada una de las capas tendrá una función específica de cara al funcionamiento del nodo completo.

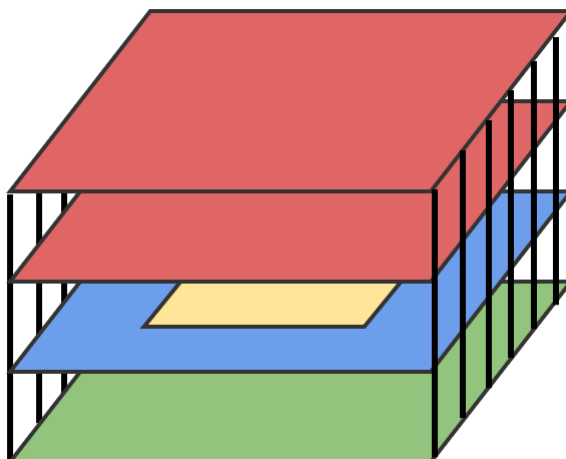


Figura 7. Diseño del nodo inalámbrico

El diseño será apilable, es decir, cada tarjeta añadida se podrá conectar encima de las ya conectadas, por ello, todas las tarjetas que se fabriquen tendrán el mismo tamaño. En la Figura 7 puede observarse el esquema de conexión de las tarjetas en nuestro nodo. En verde se encuentra la capa de alimentación, en azul la de adaptación, en amarillo la de control y comunicaciones, y en rojo la de sensado, todas ellas conectadas por pines de interconexión (líneas negras verticales).

3.2.1. Capa de alimentación

Se utilizará para proporcionar a los componentes la energía necesaria para funcionar. Para permitir un uso cómodo y mayor aceptación, se emplearán baterías para la alimentación, evitando así el uso de cableado.

3.2.2. Capa de comunicación y control

Cada nodo deberá contar con una unidad de procesamiento y comunicación, de forma que pueda comandar a los sensores, recoger sus datos y enviarlos de forma inalámbrica.

Puesto que los mayores consumos de esta clase de sistemas se producen al realizar comunicaciones, es conveniente trabajar con las señales de los sensores en cada nodo antes de enviarlas, de forma que los datos a enviar sean mínimos.

De cara al diseño de esta capa, habrá que tener en cuenta los nodos inalámbricos ya existentes en el mercado, debiendo elegir si utilizar alguno de ellos, o bien crear uno completo.

3.2.3. Capa de adaptación

En caso de utilizar un nodo para comunicaciones y control ya existente, será necesario crear una capa de adaptación que permita adaptar dicho nodo a nuestro diseño. Mediante esta capa, se podrían propagar las señales necesarias desde el nodo elegido, a los pines de nuestras propias tarjetas

4. Elección de componentes y diseño de tarjetas

Siguiendo las especificaciones que se han presentado en el capítulo anterior de cara al diseño general del sistema, habrá que elegir los componentes necesarios para cumplir los requisitos que debe satisfacer cada placa. También habrá que tomar la decisión sobre si sería más conveniente desarrollar un nodo de comunicación y control, o bien utilizar uno ya existente.

A la hora de elegir los distintos componentes habrá que tener en cuenta tanto su consumo, como su precio y su tamaño. Una vez elegidos los componentes principales (sensores, reguladores de tensión...) además habrá que elegir otros componentes secundarios cuyo uso sea necesario (condensadores de desacoplo, resistencias de *pull-up* o *pull-down*, etc.), podrán crearse esquemáticos del sistema, y *layouts* a partir de ellos. Una vez creados los *layouts* podrán fabricarse las tarjetas y soldar los componentes, pudiendo finalmente realizar las pruebas necesarias sobre éstas para comprobar su correcto funcionamiento.

4.1. Nodo de comunicaciones y control

Existen gran cantidad de nodos desarrollados previamente que podrían servir para llevar a cabo nuestro proyecto. Antes de comenzar a diseñar un nodo propio, se ha llevado a cabo una investigación sobre los modelos ya existentes, comprobando si alguno de ellos era válido para el sistema.

Si bien el número de nodos es muy amplio, no todos ellos han sido comercializados. En su gran mayoría, han sido creados por parte de universidades para realizar tareas de investigación con ellos, sin llegar a ponerse a la venta.

Analizando los distintos modelos encontrados, cabe destacar cuatro de ellos que parecían especialmente atractivos para nuestro sistema:

- **WiSMote:** Tarjeta de tamaño muy pequeño y precio razonable, que podría servir para el diseño. Sin embargo no ha sido posible obtener información sobre si se cumplen los estándares necesarios para comercializarla en Europa ni en EEUU.



Figura 8. WiSMote (www.argosystems.com)

- **WiSense:** De nuevo, la tarjeta es pequeña, y utiliza un diseño modular que permite reducir mucho el tamaño del dispositivo completo, sin embargo se trata de un producto que solo se comercializa en la India, y por tanto no cumple los estándares necesarios.



Figura 9. WiSense (www.wisense.in)

- **MTM-CM3000-MSP:** Cumple los estándares necesarios para ser comercializada tanto en Europa como en los Estados Unidos, sin embargo el tamaño es excesivo (72,80 mm x 38,50 mm).



Figura 10. MTM-CM3000-MSP (www.advanticsys.com)

- **PanStamp NRG 2:** No posee las certificaciones para los estándares, sin embargo cumple todos los requisitos definidos en ellas, y está en proceso de conseguirlas. El tamaño es realmente reducido (2,2 cm x 1,6 cm) y el precio muy asequible.

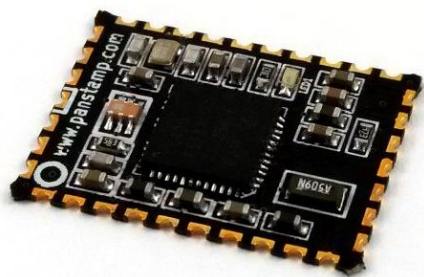


Figura 11. PanStamp NRG 2 (www.panstamp.com)

Tras analizar los distintos modelos, se llega a la conclusión de que el modelo panStamp NRG 2 cumple las necesidades del sistema a diseñar, y por tanto, en lugar de crear el nodo inalámbrico desde cero, se utilizará ese. Habrá por tanto que realizar el diseño de una tarjeta de adaptación que nos permita utilizar dicho sistema.

La tarjeta en cuestión está basada en el microcontrolador CC430F5137 de Texas Instruments, que se trata de un dispositivo de 16 bits de consumo ultra bajo, que además incluye el módulo de radio CC1101. Tiene un consumo en modo *standby* de 2 μ A. Su velocidad de procesamiento se puede programar entre 8 y 24 MHz.

Además, existen varias versiones de la panStamp NRG 2, de forma que puede hacerse funcionar en las bandas de frecuencia de 868/905/915/918 MHz. Admite alimentaciones de entre 2 y 3,6 V, y tiene un consumo de 1-2 μ A en modo *standby* y 36 mA máximo en modo transmisión de datos. Permite la programación de la tarjeta mediante SBW, BSL serie e incluso de forma inalámbrica, mediante uso del protocolo SWAP.

Este protocolo nos va a permitir comunicar los nodos entre sí. Se trata de un protocolo de comunicación inalámbrica muy simple, que envía los datos sin conocer para que son utilizados. Para implementar dicha funcionalidad, se emplean ficheros XML de descripción de dispositivos, que tienen que ser creados para cada uno de ellos.

4.2. Diseño de tarjetas

El diseño a crear será modular, y las tarjetas tendrán en general el mismo tamaño. Todas ellas serán apilables, por tanto, será necesario utilizar tiras de pines de longitud mayor a la habitual, ya que deberán atravesar la tarjeta en la que se encuentran, y conectarse a la siguiente. El *pitch* elegido para los pines será de 2,54 mm.

Si bien en un primer momento se decide utilizar un *pinout* que respete el de las tarjetas panStamp, y que es el que se ha utilizado para la primera versión de las tarjetas (y el que, por tanto, se mostrará de aquí en adelante en la memoria), wwmás tarde se decide cambiar este *pinout*, de forma que pueda reducirse el tamaño total del diseño. Las tarjetas finales contarán con los siguientes pines:

- Alimentaciones: 4 pines, voltaje de la pila, voltajes de 5 y 3,3 V y tierra.
- I2C: 2 pines, SDA y SCL
- SPI: 3 pines, SCK, MISO y MOSI
- Pin digital para desactivar las tarjetas de sensores cuando no se estén utilizando (y reducir por tanto el consumo)
- Pin para la señal de antena

Por último, se deja abierta la posibilidad de utilizar una arquitectura del sistema con tarjetas en paralelo, de forma que pudiesen colocarse las tarjetas en dos columnas al mismo tiempo. Dicha opción se describirá en más detalle al describir la tarjeta de adaptación en la sección 4.2.2.

4.2.1. Tarjeta de alimentación

Para no aumentar el tamaño del diseño se decide utilizar pilas de botón. Éstas pueden conectarse tanto en serie como en paralelo, lo que nos permitirá conseguir un voltaje mayor que el que daría una sola, y aumentar la autonomía del sistema, respectivamente.

Nuestro nodo inalámbrico funciona a 3,3 V, mientras que el sensor clave del diseño (Omron D6T) demanda una alimentación de 5 V. Por tanto, al menos estas dos alimentaciones tendrán que proporcionarse al sistema a partir de la alimentación principal que proporcionen las pilas. Para conseguir dichos voltajes de alimentación y además aumentar el tiempo de vida del sistema, se utilizarán dos pilas en serie. De este modo se consigue un total de 6 V que, aunque se vaya reduciendo a medida que el sistema consume energía, permitirá que éste siga funcionando por más tiempo.

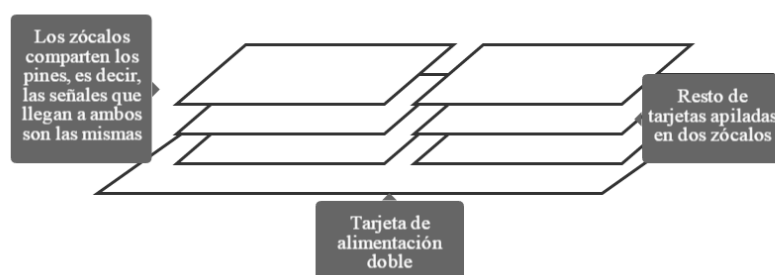


Figura 12. Utilización de la tarjeta de alimentación doble

Además, se decide crear dos modelos distintos para la tarjeta de alimentación. Por una parte tenemos el que sigue el diseño estándar, del mismo tamaño que el resto de tarjetas, y por

otro uno más grande cuyo esquema puede observarse en la Figura 12, que nos va a permitir apilar el resto de tarjetas en dos columnas distintas. De esta forma conseguiremos una mayor autonomía del sistema al incluir el doble de pilas, a la vez que se logra que, en caso de tener que utilizar muchas tarjetas, el tamaño del sistema completo no aumente tanto en una sola dimensión.

4.2.1.1. Pila

La pila a utilizar deberá tener un tamaño lo suficientemente pequeño como para que sea aceptable para el tamaño de las tarjetas a desarrollar, y su voltaje deberá cumplir con los requisitos de alimentación del sistema. Con estos parámetros fijos, habrá que elegir aquella cuya capacidad sea mayor (de forma que se maximice la autonomía del sistema).

Para cumplir dichos requisitos se decide utilizar una pila tipo RS16. Tienen un diámetro de 16 mm y una carga de 140 mA/h.

4.2.1.2. Soporte para pila

Dentro de los soportes para este tipo de pilas podemos encontrar gran variedad de modelos, si bien la mayoría aportan lo mismo. En este caso se ha elegido utilizar uno diferente de la norma.

Se empleará un componente de Keystone Electronics, con referencia 502-3. En su hoja de datos se indica que pueden utilizarse dos baterías al mismo tiempo, si bien, debido al modelo de pila elegida, solo es posible incluir una. En la Figura 13 puede observarse el modelo elegido.



Figura 13. Soporte para pila

4.2.2. Tarjeta de adaptación

Puesto que se ha decidido utilizar un nodo de comunicaciones ya existente, se hace necesario el diseño y creación de una tarjeta que adapte este nodo al resto de tarjetas diseñadas. En esta tarjeta se propagarán las distintas señales elegidas anteriormente para el *pinout* desde la tarjeta panStamp hasta las nuestras. Para ello será necesario utilizar dos tipos diferentes de conectores.

Por una parte se utilizarán los pines Arduino que se introdujeron al comienzo del capítulo 4.2. Para conectar la tarjeta panStamp a nuestras tarjetas se optará por soldar directamente dicho dispositivo a los *pads* correspondientes, sin necesidad de utilizar conectores.

Para poder trabajar con las tarjetas sin necesidad de tener baterías conectadas, se añade un puerto USB así como un traductor USB-UART que va a permitir programar la panStamp y trabajar con ella sin más que conectarla a un PC.

También se situarán en esta tarjeta los reguladores de voltaje, de modo que el sistema completo pueda alimentarse simplemente conectándose a un PC sin necesidad de conectar la

tarjeta de alimentación y sin que haya por tanto consumo de energía de las baterías, facilitando además el comando y la configuración del microcontrolador.

Por tanto, la tarjeta contendrá los siguientes componentes (además de los conectores):

- Conector USB: Para la comunicación y alimentación mediante USB de la tarjeta, es necesario incluir un conector, habrá que elegir el tipo de conector a utilizar de entre los disponibles.
- Traductor USB-UART: Para poder programar el dispositivo fácilmente, se hace necesario incluir un traductor entre la interfaz USB que utiliza el ordenador, y la interfaz UART que utiliza la tarjeta panStamp.
- Sistema de elección de alimentación: el sistema debe poder ser alimentado tanto por USB como mediante baterías. En caso de que el USB esté conectado, la alimentación por baterías debería dejar de funcionar. Será necesario implementar un sistema con dicha funcionalidad.
- Sistema de comprobación de voltaje de alimentación: para saber los niveles de las baterías y cuándo va a ser necesario tomar medidas para su reemplazo, será necesario monitorizar la tensión que proporcionan estas.
- Reguladores de alimentación: la alimentación principal del sistema, ya sea USB o mediante baterías, debe convertirse a los voltajes necesarios para alimentar los componentes, que en principio serán 3,3 V o 5 V, por tanto deberán incluirse reguladores capaces de generar dichas magnitudes.
- Antena: la tarjeta panStamp puede funcionar simplemente utilizando un cable como antena, pero puesto que la frecuencia de trabajo es de 868 MHz, este cable es de un tamaño considerable, y al ser un producto comercial, la impresión que éste da no es la mejor, por tanto deberá estudiarse la posibilidad de añadir algún otro tipo de antena (en caso de ser de otro tipo, se incluiría en una tarjeta distinta). Aun así, se incluye un conector en la tarjeta de adaptación que nos permita conectar el cable que actúe como antena.

A continuación se detallará la elección de componentes realizada para esta tarjeta:

4.2.2.1. Conector USB

Para poder trabajar de forma cómoda con el dispositivo, así como para poder alimentarlo sin necesidad de utilizar baterías, se incluye un conector USB en la tarjeta de adaptación.

Dentro de los conectores USB encontramos varios tipos distintos: conectores tipo A, tipo B, y dentro de éstos, los tipos A y B tradicionales, o bien los modelos mini y micro. En la Figura 14 se muestra una representación de los diferentes modelos.

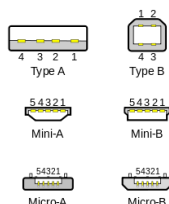


Figura 14. Tipos de conector USB

Se decide utilizar el tipo micro-B(Figura 15), ya que es un modelo muy extendido el cual utilizan en este momento la mayor parte de los dispositivos móviles. Además su tamaño es de los más reducidos.



Figura 15. Conector micro-B

Dentro de los conectores micro-B no hay una gran diferencia entre unos conectores y otros, por tanto se decide usar el modelo más económico, cuya referencia es 10118194-0001LF.

4.2.2.2. Traductor USB-UART

El método más sencillo para programar la tarjeta panStamp es mediante el uso de UART. Cualquier tipo de conversor USB-UART puede utilizarse.

Dentro de este tipo de componentes se analizan los siguientes modelos, cuyas principales características se resumen en la siguiente tabla.

USB-UART	Consumo típico (μ A)	Consumo máximo (μ A)	Precio (€)
MCP2221	46	46	2,25
CY7C64225	3	25	2,83
DS_FT230X	125	125	1,78
DS_FT231X	125	125	2,17
CY7C65213	5	5	2,45

Tabla 2. Modelos USB-UART considerados

Se decide utilizar el CY7C65213, ya que resulta el más eficiente en cuanto a consumo. Dicho chip tiene dos encapsulados distintos, por una parte está el QFN, y por otra el SSOP. Se decide utilizar el último, ya que el QFN no podría soldarse a mano.

4.2.2.3 Elección de alimentación

Puesto que el sistema debe poder alimentarse tanto mediante baterías, como mediante el uso del conector USB ya elegido, debe diseñarse un sistema que permita elegir el tipo de alimentación que va a utilizarse en un momento determinado.

Puesto que en caso de que el cable USB se encuentre conectado podrá utilizarse directamente su alimentación, se decide utilizar el circuito mostrado en la Figura 16 para tal tarea.

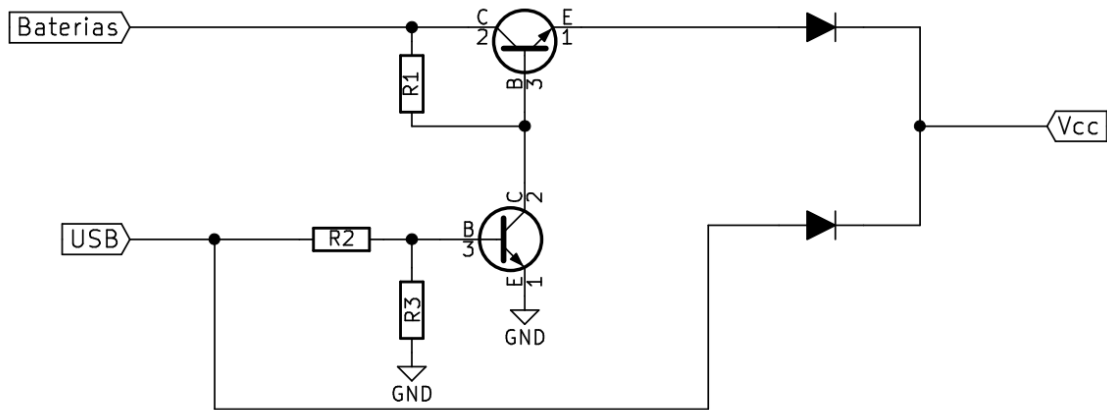


Figura 16. Sistema de elección de alimentación

Dicho circuito actúa del siguiente modo: cuando el USB no proporciona alimentación, el transistor inferior no permite pasar la corriente, de forma que toda la corriente de la batería atraviesa el transistor superior, y será esta la que alimente los componentes. Sin embargo, si el USB está conectado, el transistor inferior dejará pasar la corriente de la pila (eligiendo un valor lo suficientemente grande de R1 se conseguirá que el consumo no sea demasiado alto), y por tanto será el voltaje proporcionado por el USB el que alimente el sistema.

4.2.2.4. Comprobación de voltaje de alimentación

Puesto que el principal modo de funcionamiento del sistema va a ser mediante baterías, es necesario implementar un sistema que permita comprobar el nivel de éstas, de forma que sea posible saber cuándo va a ser necesario llevar a cabo un mantenimiento del sistema.

Para esta tarea, se decide utilizar simplemente un divisor resistivo, cuyo circuito puede contemplarse en la Figura 17. El valor de las resistencias se elige considerablemente alto para que el consumo no sea excesivo, ya que el consumo de esta parte del diseño será continuo.

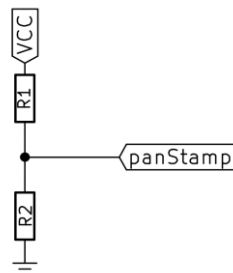


Figura 17. Divisor resistivo

El valor elegido para las resistencias será de 10 MΩ para ambas. El divisor resistivo sigue las siguientes ecuaciones:

$$V_{out} = \frac{V_{CC}}{R_1 + R_2} * R_2 \quad (1)$$

Por tanto, para nuestro valor de entrada máximo de 6 V obtendremos 3 V en el divisor, algo por debajo del nivel máximo que permite la tarjeta panStamp a su entrada:

$$V_{out} = \frac{6}{10 \text{ M} + 10 \text{ M}} * 10 \text{ M} = 3 \text{ V} \quad (2)$$

4.2.2.5. Reguladores de alimentación

Las alimentaciones de entrada al sistema elegidas pueden proporcionarnos hasta 6 V, o 5 V, pero los componentes que van a utilizarse (y en general la mayoría de componentes comerciales) tienden a utilizar tensiones de alimentación de 3,3 V o 5 V. Adicionalmente, las baterías se degradan con su uso, por tanto, para convertir estas tensiones a los valores necesarios, así como para garantizar un voltaje estable se han incluido reguladores en el diseño.

Puesto que tenemos dos valores de salida, cabe la posibilidad de que haya que usar dos reguladores, o también es posible utilizar solo uno que tenga un rango de salidas amplio, de forma que el mismo sirva para ambos.

Habrà que tener en cuenta también la corriente de salida que deberá ser capaz de proporcionar los reguladores. El regulador de 5 V se prevé que vaya a tener un menor uso (menos componentes utilizarán la alimentación de 5 V que la de 3,3 V), por tanto su corriente de salida máxima deberá ser menor que la del de 3,3 V. Se estima que, puesto que los consumos del sistema son considerablemente pequeños, con 150 mA para el regulador de 3,3 V y 50 mA para el de 5 V será suficiente.

Por último, a la hora de realizar las conexiones de los reguladores, podemos elegir utilizar una arquitectura serie o paralelo (es decir, podemos conectar ambos reguladores al voltaje de la pila, o bien el de 5 V al voltaje de la pila, y el de 3 V a la salida del regulador de 5 V). Dependiendo del tipo de arquitectura elegida, los parámetros de los reguladores a utilizar variarán.

Dentro de los reguladores de tensión hallamos una gran variedad. La siguiente tabla muestra una comparación entre los más relevantes:

Regulador	Alimentación mínima (V)	Alimentación máxima (V)	Consumo típico (μ A)	Consumo máximo (μ A)	Corriente máxima (mA)	Precio (€)
TPS781	2,2	5,5	1	8	230	0,84
TPS770	2,7	10	17	28	50	1,02
TC1070	2,7	6	50	80	300	0,23624
AP133	2	6	40	60	300	0,63
AP7335	2	6	35	80	300	0,37
AP7365	2	6	35	80	600	0,4
MIC5225	2,3	16	29	50	150	0,79
TPS781	2,2	5,5	1	8	230	0,84
TPS715	2,5	24	3,2	5,8	50	0,3
MIC5235	2,3	24	18	35	150	0,92
STLQ50	2,3	12	3,5	5	50	0,66
TPS780	2,2	5,5	0,42	5	150	0,98
TPS799	2,7	6,5	40	60	200	0,86
TPS71501	2,5	24	3,2	12	50	0,78
TPS76201	2,7	10	23	30	100	1,21

Tabla 3. Reguladores de tensión considerados

Tras analizar las diferentes opciones, se decide utilizar la arquitectura en paralelo, lo que nos va a permitir disminuir los consumos enormemente al poder utilizar un regulador para la alimentación de 5 V que pueda proporcionar menor nivel de corriente a su salida.

Para obtener la tensión de 3,3 V, se busca una corriente de salida de al menos 150 mA. Dentro de las opciones presentadas, el regulador que mejor se adapta es el MIC5235, ya que cumple con todos los requisitos del diseño, y su consumo es reducido.

Para la alimentación de 5 V, la corriente de salida necesaria es de al menos 50 mA. Se elige utilizar el STLQ50, ya que es el que, adaptándose a todos los requisitos necesarios, tiene un menor consumo.

4.2.3. Tarjeta de sensores

Para conseguir facilidad de uso del sistema, de forma que puedan intercambiarse las tarjetas de sensores que se están utilizando de forma sencilla, se decide incluir en cada tarjeta un sistema de identificación (una clave de 8 bits), de forma que el microcontrolador pueda saber qué tipo de dispositivos tiene conectados en cada momento.

Si bien en principio pensaba utilizarse una clave de 16 bits, finalmente decide incluirse una de tan solo 8 bits, logrando así un tamaño más reducido. Mediante esta clave, el microcontrolador de cada nodo podrá saber qué clase de tarjetas tiene conectadas, y por tanto, qué medidas puede realizar. De los ocho bits, se decide fijar cuatro para identificar el tipo de tarjeta, y utilizar los otros cuatro para identificar la tarjeta específica (esto va a permitir incluir varias tarjetas de sensores iguales trabajando en el mismo nodo).

Para implementar dicha clave se utilizará un expensor entrada/salida, que traducirá los 8 bits a I2C para poder comunicarse con el microcontrolador. Se añadirán también interruptores de carga que permitan desactivar la alimentación de los componentes de la tarjeta cuando no sean necesarios, permitiendo así ahorrar energía.

Para el primer prototipo se incluyen los sensores que se consideran más necesarios para el diseño: un sensor de temperatura, uno de humedad y el sensor CCD Omron D6T.

A continuación se detallará la elección de componentes que se ha llevado a cabo para el diseño de las tarjetas de sensores.

4.2.3.1. Interruptores

Se busca utilizar interruptores lo más pequeños posibles, de forma que no se ocupe gran espacio en la tarjeta para esta funcionalidad. De entre los modelos disponibles (entre los que no existen grandes diferencias técnicas), se elige utilizar el A6H de Omron por ser el más pequeño.

4.2.3.2. Expansor entrada/salida

Para elegir el expensor entrada/salida se sigue el procedimiento que se ha seguido para el resto de componentes, se busca aquel que cumpliendo los requisitos básicos (número de entradas y voltajes de alimentación) presente el menor consumo, tamaño y precio.

Expansor	Consumo típico (μ A)	Consumo máximo (μ A)	Precio (€)
CAT9555	135	200	1,59
PCA6416A	47,5	100	1,6
PCA9535A	47,5	100	1,6
PCA9555	135	200	1,46
PCA9555A	47,5	100	1,6
PCA9535	30	75	1,6
PCA9557	19	25	1,34

Tabla 4. Modelos de los expansores entrada/salida considerados

La primera opción consistía en realizar el sistema de identificación de 16 bits, de esta forma se conseguiría poder utilizar gran cantidad de tarjetas diferentes al mismo tiempo, pero puesto que el tamaño que ocupaba era excesivo, se decide finalmente utilizar menor cantidad de bits. Por este motivo, la comparativa mostrada en la Tabla 4 corresponde a los expansores de 16 bits. A la hora de elegir el que finalmente se va a usar, se eligió el modelo PCA9557 perteneciente a la misma familia pero con menor número de entradas, que presentaba las mismas prestaciones que los de la tabla anterior con un consumo mínimo incluso menor.

4.2.3.3. Sensor térmico Omron

El punto clave del presente diseño se basa en el uso del sensor térmico de Omron D6T-44L-06. Este sensor actúa como una cámara que, en lugar de enviar imágenes, envía la temperatura de un determinado píxel.

El sensor elegido, posee una cuadrícula de 4 x 4 píxeles. El sensor medirá y devolverá la temperatura de cada una de las zonas cubiertas por esta cuadrícula, asignándolas una temperatura. De esta forma, si se sitúa el sensor en la posición correcta (como apuntando hacia una cama), podremos saber cuándo se dan situaciones anómalas (la persona se ha levantado de la cama a mitad de la noche pero no ha vuelto a acostarse, o sigue en la cama a una hora poco normal...), todo ello sin vulnerar en ningún momento la intimidad de las personas monitorizadas ya que no se envía una imagen como tal.

Los datos que envía el sensor son medidas de la temperatura superficial que detecta el sensor en los objetos que tiene delante. En la Figura 18 puede observarse una representación de los datos que toma el sensor. El color azul representa la temperatura ambiente y el rojo el resto. Idealmente el sensor detectaría la superficie ocupada por la persona completa, si bien en la práctica no es lo normal, ya que la ropa mitiga enormemente la temperatura captada por éste, de forma que generalmente, el sensor solo detectaría los píxeles relativos a la cabeza, como se muestra en la figura.

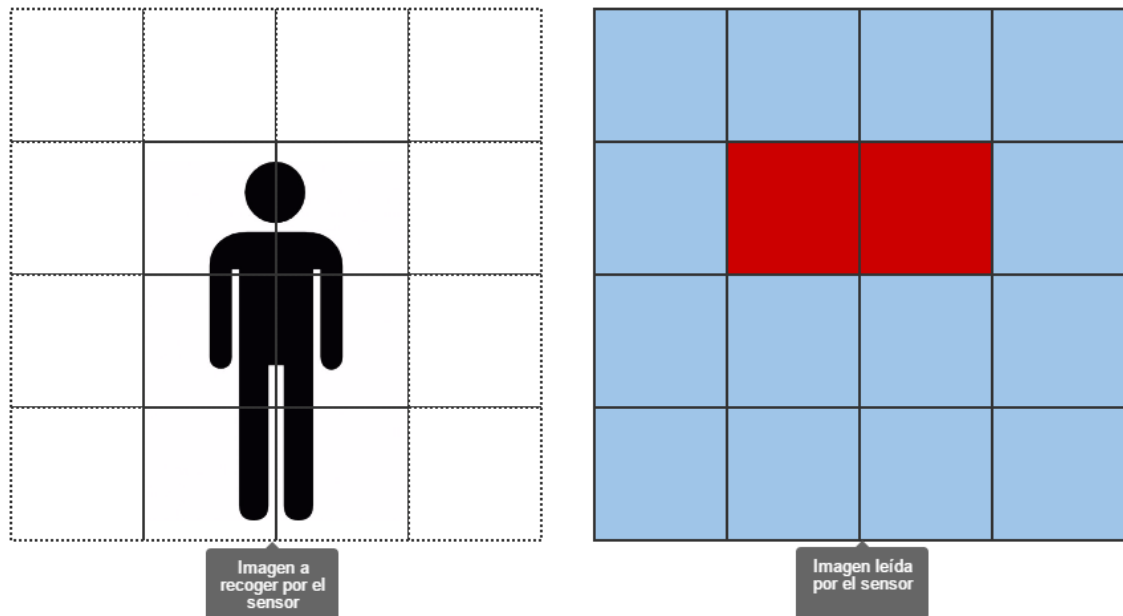


Figura 18. Esquema de lectura del CCD

Este sensor, que se muestra en la Figura 19, necesita una alimentación de 5 V y se comunica mediante I2C. Puesto que nuestro microcontrolador funciona a 3,3 V será necesario introducir un convertidor de nivel I2C, que pasará de los 5 V del sensor a 3,3 V y viceversa.



Figura 19. Sensor Omron D6T

Además, el sensor no posee los medios para incluirlo directamente en la tarjeta, por tanto habrá que buscar también un conector macho que nos permita conectarlo.

El conector a utilizar será el recomendado en la propia hoja de datos del sensor. Se buscará además un modelo horizontal, para poder colocar el sensor en paralelo a las tarjetas desarrolladas.

4.2.3.4. Convertidor de nivel para comunicaciones I2C

Las comunicaciones I2C de nuestro microcontrolador funcionan a 3,3 V, sin embargo el sensor Omron funciona con una alimentación de 5 V. Por tanto, para poder realizar comunicaciones entre el sensor y el microcontrolador, es necesario introducir un dispositivo que pase de un voltaje a otro.

Se opta por utilizar el modelo PCA9306, con un consumo de 5 μ A y un precio de 1,69 €.

4.2.3.5. Interruptor de carga

Para conseguir un mayor ahorro de energía, y por tanto una mayor autonomía del sistema, se incorpora en cada tarjeta de sensores un interruptor de carga por cada alimentación a utilizar, que dejará pasar las diferentes corrientes de alimentación solo cuando se van a utilizar los sensores, de este modo se reduce el consumo del sistema. Se han considerado los siguientes modelos registrados en la Tabla 5:

Interruptor de carga	Consumo típico (μ A)	Consumo máximo (μ A)	Precio (€)
AP2805	60	80	0,51
AP2822	68	95	0,46
CD00167470	40	70	0,62
2AP2161D_71D	45	70	0,45
RT9711C	25	45	0,51

Tabla 5. Modelos de interruptores de carga considerados

Se elige el RT9711C, una vez más por ser el de menor consumo.

4.2.3.6. Sensores de temperatura y humedad

Aunque el sensor Omron sea capaz de dar lecturas de temperatura, éstas están centradas en una zona determinada. Además de ese tipo de medidas, también se quiere tener una medida de la temperatura ambiente, así como de la humedad.

De cara a implementar este sistema de medida debemos tener en cuenta varias posibilidades que se describen a continuación:

- Sensores de humedad y temperatura por separado y salida I2C.
- Sensores de humedad y temperatura en un mismo chip y salida I2C.
- Sensores de humedad y temperatura por separado y salida analógica.
- Sensores de humedad y temperatura en un mismo chip y salida analógica.

Para los dos últimos casos (salida analógica), habría que tener en cuenta que sería necesario incluir expansores entrada/salida y ADCs, o bien utilizar un ADC que funcione por I2C, para traducir el valor analógico de salida a I2C de forma que no se fijaran pines extra para una tarjeta en particular.

Sensor	Tipo	Consumo típico (μA).	Consumo máximo (μA)	Comunicación	Precisión Hum. (%)	Precisión Temp. (°C)	Precio (€)
HIH6000	H/T	650	1000	I2C	±4,5	±0,5	8,75
HIH-4030/31	H	200	500	Analógico	±3,5	-	12,22
HIH-5030/31	H	200	500	Analógico	±3	-	6,89
LMT87	T	5,4	9	Analógico	-	±0,3	0,59
LMT88	T	4,5	10	Analógico	-	±1,5	0,61
MCP9700	T	6	12	Analógico	-	±4	0,26
LM75B	T	100	300	I2C	-	±2	0,59
TCN75A	T	200	500	I2C	-	±1	0,73
AT30TS74	T	400	800	I2C	-	±1	0,8
QM1H0P0073	H/T	750	1100	I2C	±2	±0.3	1,8

Tabla 6. Modelos de sensores de humedad y temperatura considerados

ADC	Consumo Típico (μA)	Consumo máximo (μA)	Precio (€)
ADC081C021	160	300	1,53
MCP3021	195	370	1,12

Tabla 7. Modelos de ADC considerados

Tras comprobar varias posibilidades, se decide utilizar un chip que incluya ambos sensores, y cuya salida además sea I2C. De este modo es posible conseguir un consumo muy similar a cualquier otro acercamiento, pero sin embargo se reduce considerablemente el tamaño demandado por los sensores y el precio.

Se elige utilizar un dispositivo de la familia HIH6000 por ser el que mejor se adapta al diseño. Adicionalmente se incluye también el QM1HP0073, ya que es considerablemente más pequeño y su precio es mucho menor, si bien su encapsulado hace muy difícil de soldar. Por tanto se añadirán ambos sensores a la tarjeta, y se soldará el que más convenga.

4.2.4. Otros componentes

Además de los chips de sensores, reguladores, expansores, etc., que se necesitan para implementar el diseño, cada uno de estos dispositivos necesita además otros componentes para funcionar correctamente, que incluyen generalmente condensadores de desacoplo y resistencias. Además, puesto que no es de especial importancia para el desarrollo del primer prototipo del sistema, se incluye en este punto también la elección de la antena.

A partir de la hoja de datos de cada uno de los componentes puede comprobarse cuáles son necesarios. En la medida de lo posible y puesto que las tarjetas son de un tamaño muy reducido, los componentes buscados deberán ser de montaje superficial. El encapsulado elegido para dichos componentes será el 0603.

Será necesario incluir:

- Condensadores de desacoplo para la mayor parte de los chips
- Resistencias de *pull-up* para las líneas I2C de 3,3 V y 5 V y los bits fijos de identificación de tarjetas
- Resistencias de *pull-down* para las salidas de los interruptores
- Otras resistencias que recomiendan los fabricantes de algunos de los chips para determinados pines
- Resistencias para el circuito de elección de alimentación
- Cable de longitud $\lambda/4$ actuando como antena

4.3. Diseño de esquemáticos

Una vez elegidos todos los componentes puede pasarse a diseñar los esquemáticos del dispositivo. Para realizar dicha tarea se hará uso del software libre KiCad en su versión 3.0.2, que nos permitirá además crear después los *layouts* a partir de estos esquemáticos.

4.3.1. Tarjeta de alimentación estándar

Esta tarjeta debe contar con el conector estándar de nuestras tarjetas, así como con un soporte para pilas:

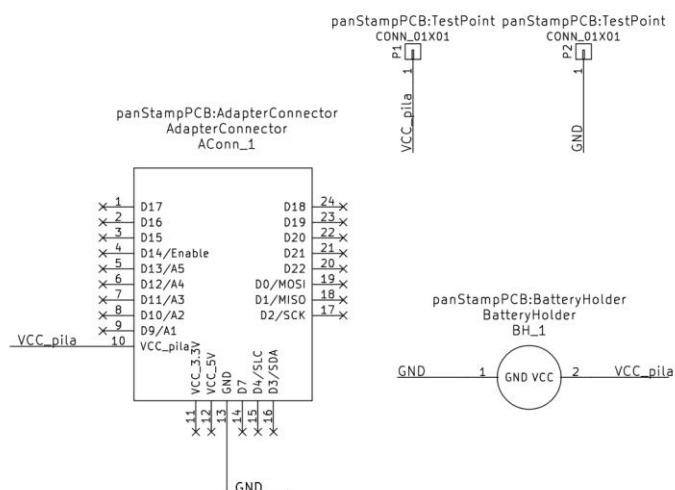


Figura 20. Esquemático de la tarjeta de alimentación estándar

4.3.2. Tarjeta de alimentación doble

Debe incluir dos conectores, y dos soportes para pilas, que se conectarán en paralelo para conseguir mayor autonomía:

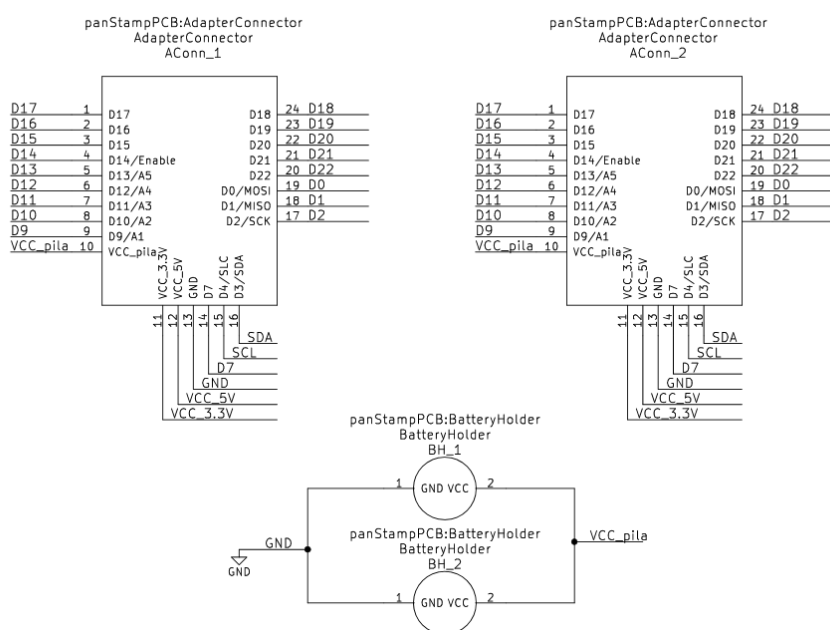


Figura 21. Esquemático de la tarjeta de alimentación doble

DISEÑO DE SISTEMAS DE SENSORES PARA TELEASISTENCIA

4.3.3. Tarjeta de adaptación

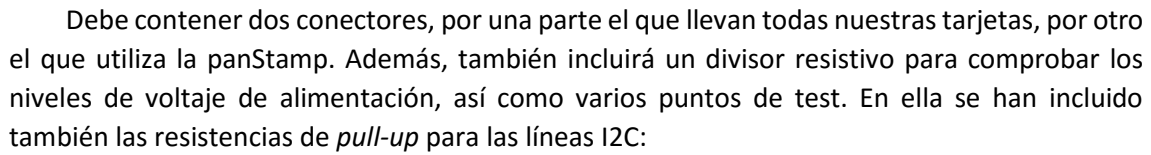


Figura 22. Conectores, antena, divisor resistivo y resistencias de pull-up de la tarjeta de adaptación

Incluimos en ella también el sistema de elección de alimentación:

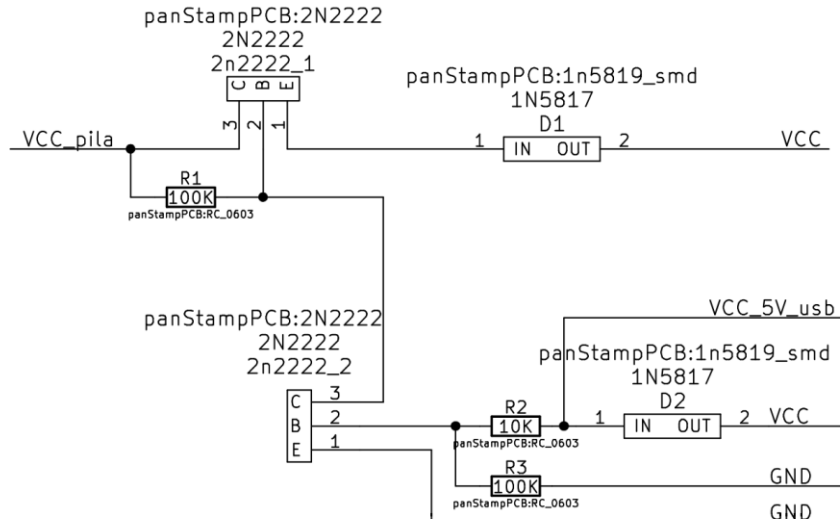
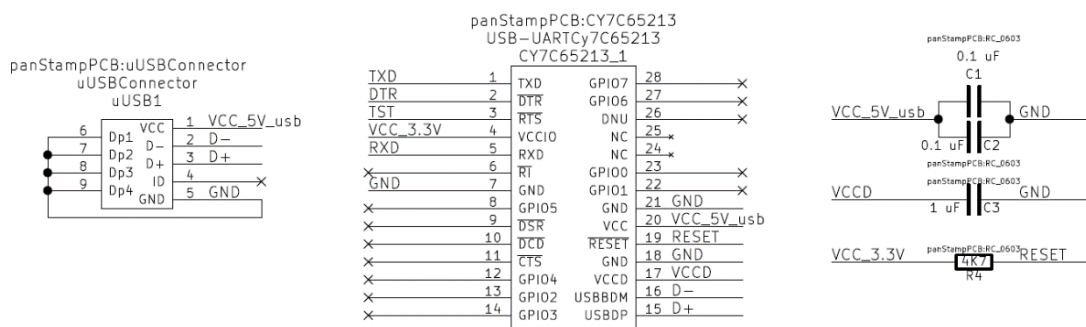


Figura 23. Esquemático del sistema de elección de alimentación

Para evitar tener que utilizar pines extra del conector, incluimos en ella también el conector micro-USB y el traductor USB-UART:



Además, deberá tener los interruptores de carga que permitan cortar la alimentación al resto de componentes para ahorrar energía:

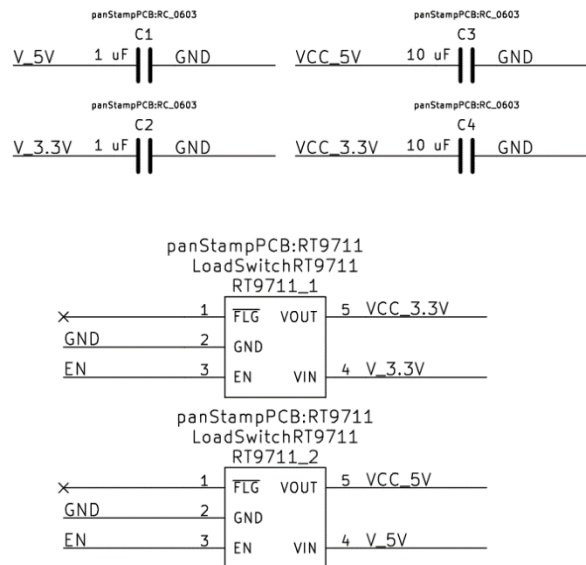


Figura 27. Esquemático de los interruptores de carga de la tarjeta de sensores

El conector para el sensor Omron junto con el convertidor de nivel I2C:

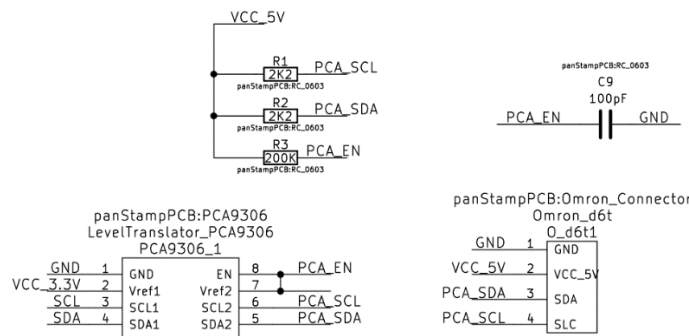


Figura 28. Esquemático del conector para el sensor Omron D6T junto al convertidor de nivel I2C

También debe incluirse el sistema de identificación de la tarjeta, formado por los interruptores, el expansor, y las resistencias de *pull-up/pull-down* para los bits fijos:

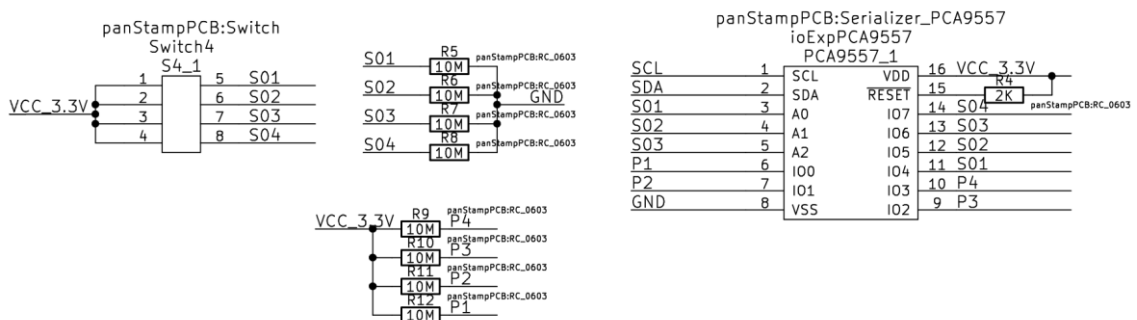


Figura 29. Esquemático del sistema de identificación de tarjetas

Por último, incluye los dos sensores de temperatura y humedad:

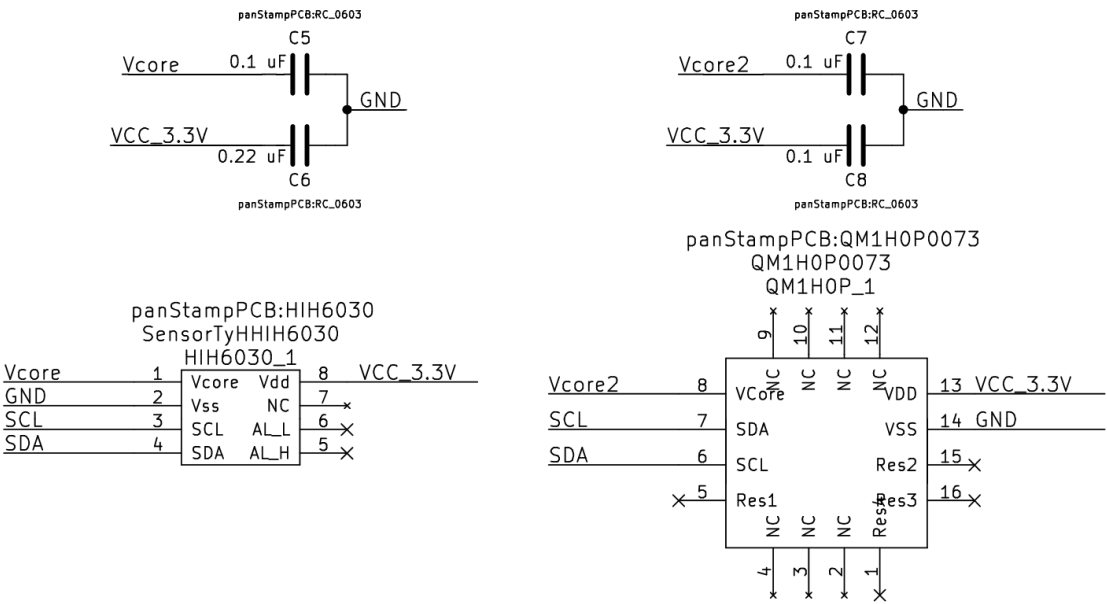


Figura 30. Esquemático de los sensores de humedad y temperatura

4.4. Diseño de *layouts*

Para llevar a cabo el diseño de los *layouts*, hay que tener en cuenta que se pretende que las tarjetas sean de tamaño reducido, por tanto habrá que tratar de juntar los componentes tanto como sea posible, y utilizar anchos de pista y separación entre pistas bajos.

Puesto que los valores de corriente atravesando las pistas serán bastante bajos (bastante por debajo de los 150 mA generalmente), los tamaños de pista pueden ser bastante pequeños también. Aun así, habrá que tener en cuenta que las soldaduras se realizarán a mano, y por tanto será necesario que las vías sean algo más grandes.

También se tratará de evitar situar las vías bajo componentes, ya que puede resultar difícil realizar la soldadura de la vía de forma que no entorpezca la del componente.

En la medida de lo posible, se tratará de realizar la mayor parte de colocación de componentes y rutado en la misma capa de la tarjeta, de forma que sea posible situar un plano de masa en la contraria, que de mayor estabilidad al diseño.

El tamaño final de las tarjetas queda definido por el tamaño necesario para incluir todos los componentes y rutado de la tarjeta de adaptación, que resulta ser de 32 x 32 mm.

Por petición de la empresa para la que se ha llevado a cabo el diseño, no se incluirá mayor información sobre la creación de los *layouts* realizada.

5. Desarrollo Software

En este capítulo se mostrará cómo se ha llevado a cabo el desarrollo de los programas de software utilizados tanto para realizar el control de cada nodo (para lo que se usará Arduino, compatible con las tarjetas panStamp), como para realizar la monitorización de la red de sensores completa (para esta tarea se utilizará el software Netbeans y el lenguaje de programación Java).

El código desarrollado para poner en funcionamiento el sistema no se incluirá en la presente memoria por petición de la empresa para la que se ha desarrollado el dispositivo.

5.1. Control del microcontrolador

Para realizar el control del microcontrolador se creará un programa en Arduino, que es compatible con las tarjetas panStamp. Para ello, para crear el programa definitivo que controle los nodos y se encargue de la comunicación entre ellos, se cuenta por una parte con la propia API de panStamp, que proporciona funciones que permiten controlar a bajo nivel el intercambio de paquetes inalámbricos, y por otra parte con el protocolo SWAP, que permite abstraerse de los detalles de bajo nivel, pero que se considera innecesario, al menos para la primera versión del sistema.

5.1.1. Comunicación con los sensores

En primer lugar, y de forma que pueda comprobarse el correcto funcionamiento de las tarjetas diseñadas, se escriben varios programas que comprueben que la comunicación con cada uno de los sensores mediante I2C puede realizarse de forma correcta.

Cada uno de los programas tendrá partes de código comunes, que se detallarán a continuación.

5.1.1.1. Código Común

En cada uno de los programas se incluirán las librerías:

- `Arduino.h`: Librería básica de Arduino
- `Wire.h`: Para comunicaciones I2C
- `HardwareSerial.h`: Para permitir utilizar el puerto serie para comunicación con el PC

Además, en la etapa de inicialización todos utilizarán los mismos comandos:

```
void setup() {  
  Wire.begin();           // Join i2c bus (address optional for master)  
  Serial.begin(38400);    // Start serial for output  
  pinMode(14,OUTPUT);     // Select the activation pin as an OUTPUT  
  digitalWrite(14,HIGH);  // Activate the enable signal  
}
```

5.1.1.2. Sensor HIH6030

Primero se trata de crear un programa que permita comunicar la tarjeta panStamp con el dispositivo HIH6030. Para ello es necesario conocer la dirección del dispositivo (0x27) así como cuál es el proceso que hay que seguir para comunicarse con él. Ambos datos pueden encontrarse en su hoja de datos, y se muestran en la Figura 31:

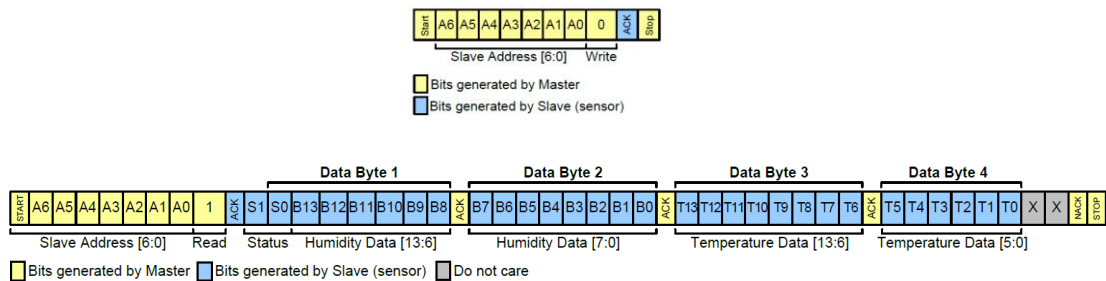


Figura 31. Esquema de comunicaciones HIH6030

Conociendo estos datos, pueden realizarse las comunicaciones necesarias con el dispositivo. En primer lugar se despertará al sensor mediante el envío de su dirección (0x27) y el bit de escritura, para conseguir dicha funcionalidad se utiliza la función `Wire.beginTransmission(address)`.

Tras esto, habría que cerrar la comunicación, y comenzar una nueva, esta vez dirigida a su dirección de lectura. Esto en Arduino se consigue mediante el uso de `Wire.endTransmission()` y `Wire.requestFrom(dirección, bytes)`:

```
Wire.beginTransmission(0x27);           // Begin communication, write mode
Wire.endTransmission();                 // End communication
result = Wire.requestFrom(0x27, 4);    // Request 4 bytes
```

La función `Wire.requestFrom()` devuelve tras su ejecución un número, que representa los bytes recibidos. Con esto podemos ver si realmente hemos podido comunicarnos satisfactoriamente con el dispositivo. Tras leer los datos, se cambia su formato para poder entenderlo fácilmente y se muestran por pantalla:

```
if(result > 0){                          // If data is received
    int cont = 0;
    while (Wire.available()) {           // Slave may send less than requested
        data[cont] = Wire.read();        // Read the data received, 1 byte at time
        cont++;
    }
}
else{                                    // If no data was received
    Serial.println(String("No data was received..."));
}
data[0] = data[0] & 0x3F;                // Apply the masks needed
data[1] = data[1] & 0xFF;
data[2] = data[2] & 0xFF;
data[3] = data[3] & 0xFC;
temperature = (data[2] << 6) | (data[3] >> 2);
humidity = (data[0] << 8) | (data[1]);
// Calculate the temperature and humidity
temp = ((temperature / (pow(2, 14) - 1)) * 165) - 40;
hum = ((humidity / (pow(2, 14) - 1)) * 100);
```

5.1.1.3. Sensor OMRON D6T

De nuevo, en primer lugar hay que comprobar cómo deben realizarse las comunicaciones con el dispositivo en la hoja de datos de éste. En ella puede observarse que la dirección I2C del sensor es la 0x14, y que para comunicarse con él hay que seguir el siguiente proceso (en la hoja de datos aparece una errata, los valores mostrados como P6 y P7 en la Figura 32 deberían ser P14 y P15 respectivamente):

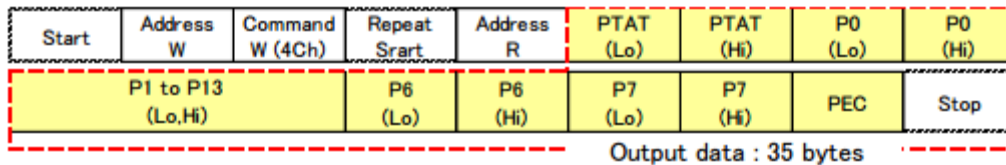


Figura 32. Esquema de comunicaciones del sensor Omron D6T

Con esto, se escribe el código empleando las mismas funciones que para el sensor anterior, y además con la función *Wire.write(data)*, que nos permite enviar un byte de datos por la línea I2C, y que utilizaremos para enviar el comando 0x4C.

```
byte address = 0x0A;
Wire.beginTransaction(address);
Wire.write(0x4C);
Wire.endTransmission(false);
result = Wire.requestFrom(address, 35);
if(result > 0){
    int cont = 0;
    while (Wire.available()) {
        data[cont] = Wire.read();
        cont++;
    }
    flag=0;
}
else{
    Serial.println(String("No data was received..."));
    flag=1;
}
if(flag==0){
    ptat = ((data[1] << 8) | data[0])/10;
    Serial.println(String("Reference temperature inside the D6T: ") + ptat);
    for(int i=2; i<34;i=i+2){
        aux = ((data[i+1] <<8) | data[i])/10;
        Serial.println(String("Showing pixel ") + (i/2) + String(" data read: ")
            + aux);
    }
}
```

5.1.1.4. Expansor entrada/salida PCA9557

Una vez más, en la hoja de datos del dispositivo podemos encontrar los datos necesarios para comunicarnos con él. En este caso, la dirección tiene fijados 4 de los bits, y nos permite fijar los 3 restantes, con lo que finalmente tenemos la siguiente dirección: 0x1F

El dispositivo consta de cuatro registros que nos van a permitir determinar el modo de funcionamiento de cada pin de E/S como entrada (registro contiene un '1') o salida (registro contiene un '0'), leer el valor de los pines (sean entradas o salidas), transmitir un determinado valor en las salidas o bien invertir la funcionalidad de alguno de los pines definidos con el primer registro descrito (si es entrada pasa a ser salida y viceversa). En la Figura 33 y la Figura 34 pueden observarse los datos proporcionados por el fabricante para el uso del dispositivo.

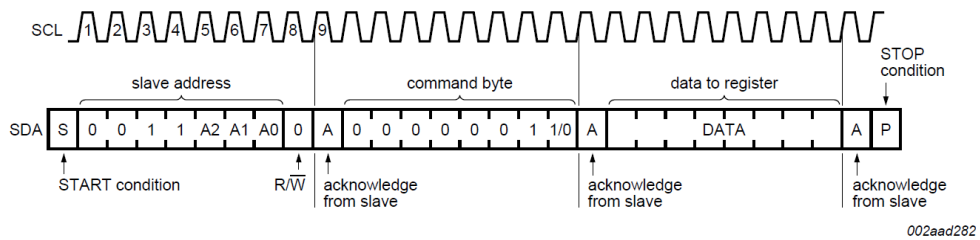


Figura 33. PCA9557: Introducir configuración de E/S

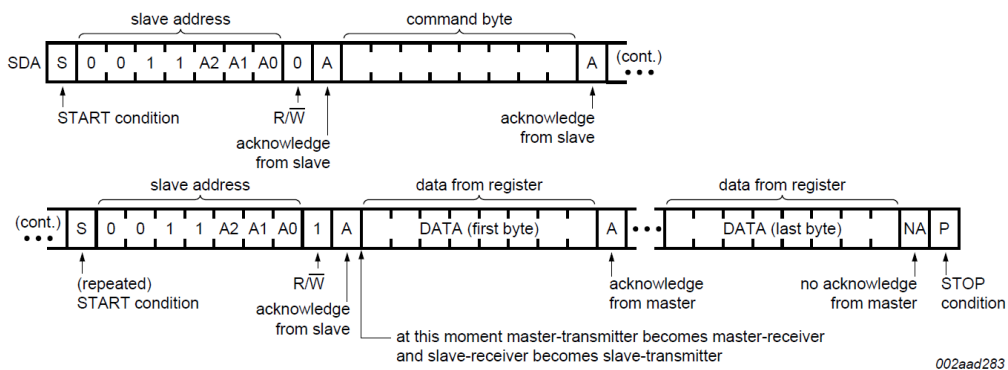


Figura 34. PCA9557: Lectura de registro

En este caso, puesto que queremos que los pines funcionen como entradas, escribiremos todos los valores a '1' en el registro correspondiente (0x03). Tras esto será necesario leer sus valores, para lo que utilizaremos el registro 0x00. También hay que tener en cuenta la configuración de polarización inicial del componente, que puede observarse en la Figura 35, por tanto, deberá escribirse el valor 0x00 en el registro 0x02.

Table 7. Register 2 - Polarity inversion register bit allocation

Bit	7	6	5	4	3	2	1	0
Symbol	N7	N6	N5	N4	N3	N2	N1	N0
Default	1	1	1	1	0	0	0	0

Figura 35. Configuración de polaridad

```
byte address = 0x1F;
Wire.beginTransaction(address);
Wire.write(0x03);           //Set all pins as inputs
Wire.write(0xFF);
Wire.endTransmission();
```



```
Wire.beginTransaction(address);  
Wire.write(0x02);           //Keep the polarity of the inputs  
Wire.write(0x00);  
Wire.endTransmission();
```

Una vez configurado el chip, puede realizarse la lectura correspondiente para obtener el valor que tiene a sus entradas:

```
Wire.beginTransaction(address);  
Wire.write(0x00);  
Wire.endTransmission();  
result = Wire.requestFrom(address, 1);  
if(result > 0){  
    int cont = 0;  
    while (Wire.available()) {  
        data = Wire.read();  
        Serial.println(String("Data read: ") + data);  
        cont++;  
    }  
}  
else{  
    Serial.println(String("No data was received"...));  
}
```

5.2. Software para interfaz y control del sistema

Para implementar el sistema mediante el cual puedan leerse las medidas tomadas por los sensores, así como controlar estos, se utilizará Java. Para ello, se desarrollará un programa dividido en distintas capas. En la Figura 36 puede observarse un diagrama UML del desarrollo realizado.

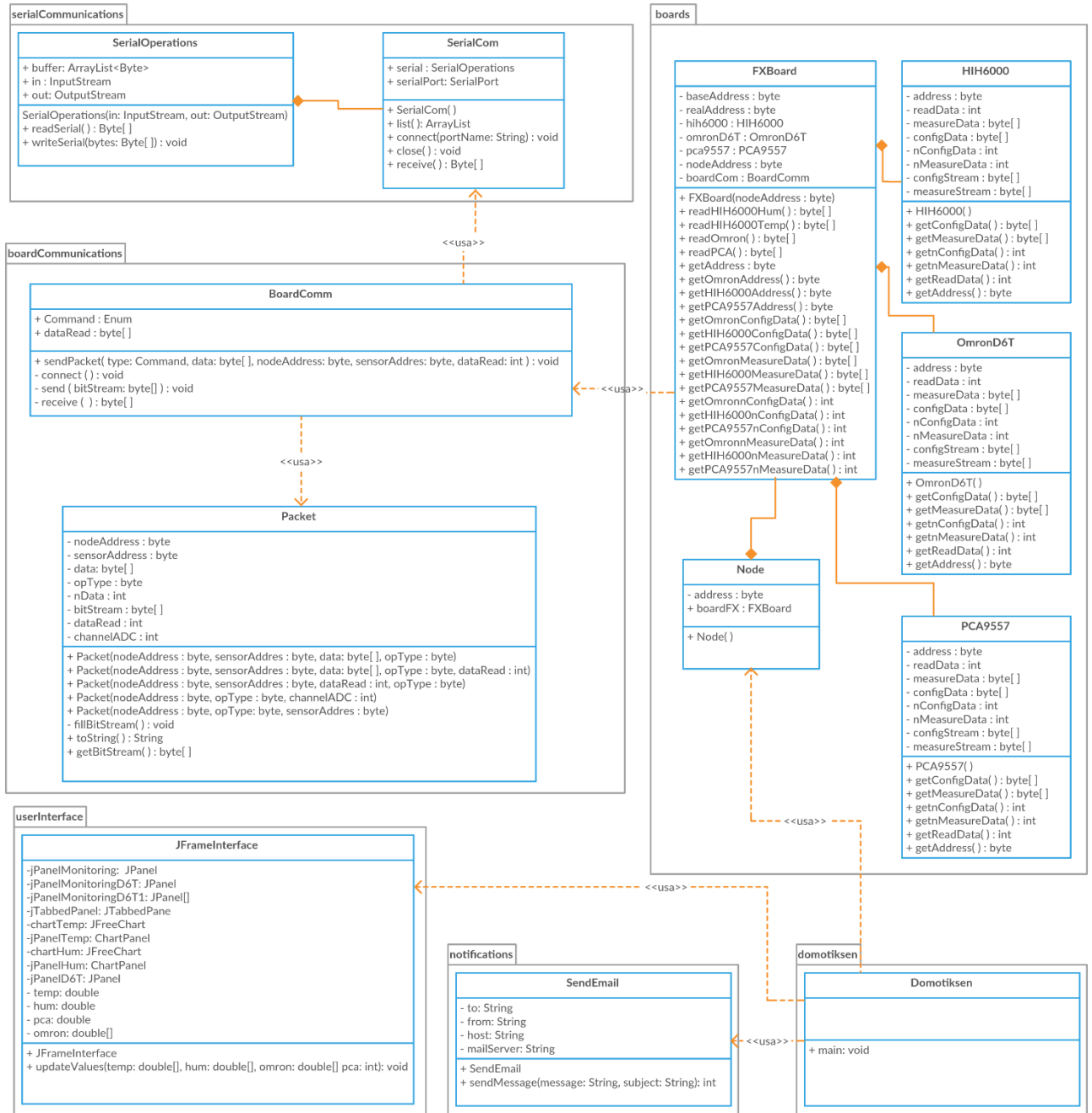


Figura 36. Diagrama UML del software desarrollado

Una primera capa de comunicación serie se encargará de enviar y recibir datos, así como de realizar la conexión con el puerto UART correspondiente (*serialCommunications*). La siguiente capa se encargará de las comunicaciones inalámbricas (*boardCommunications*). Tras la anterior, se sitúa una nueva capa encargada de la gestión de tarjetas conectadas a un nodo determinado (*boards*). Mediante una capa de adaptación (*domotiksen*) se consigue poner a trabajar los diferentes nodos del sistema, mediante el uso de las capas inferiores. Por último, se incluye una capa de usuario formada por la interfaz gráfica del programa (*graphicInterface*), y un sistema de notificación por correo electrónico (*notification*).

5.2.1. Capa de comunicación serie

Será la encargada de transmitir y recibir los mensajes necesarios por la interfaz UART, de forma que puedan enviarse y recibirse mensajes desde el programa ejecutándose para monitorización y las tarjetas de sensores.

Para desarrollar dicha capa, se ha utilizado la librería RXTX. Se trata de una librería de *Java* que proporciona comunicaciones serie y paralelo para el Java Development Toolkit.

En primer lugar, se realiza la conexión al puerto serie en el que está conectada la tarjeta correspondiente, para ello se utilizan los parámetros por defecto de las tarjetas panStamp, que son:

- Velocidad de transmisión de datos de 38400 bps
- 8 bits para datos
- Un bit de parada
- Sin paridad

Tras establecer la conexión, será posible realizar comunicaciones entre ambos dispositivos.

5.2.2. Capa de comunicación inalámbrica

Capa encargada de realizar las comunicaciones inalámbricas entre las distintas tarjetas. Para ello, en primer lugar se hace necesario elegir como funcionarán las tarjetas. Por una parte, se podría diseñar el sistema de forma que, desde el ordenador encargado de la monitorización se pida simplemente una determinada medida (temperatura, humedad, lectura del sensor Omron D6T...), y a partir del mensaje de petición sea la propia tarjeta de sensores la que se encargue de realizar las distintas operaciones, es decir, en la propia tarjeta estarían almacenados los diferentes códigos necesarios para leer y configurar los diferentes sensores. Por otra parte, se pueden configurar las tarjetas de forma que solamente realicen unas determinadas operaciones, cuyos datos se le proporcionarían de forma inalámbrica, de forma que no tengan información sobre qué están haciendo y se dediquen solamente a enviar los datos que reciban por el pin correspondiente.

Tras valorar ambas opciones, se decide utilizar la última, fundamentalmente debido a que, mediante este método podría reducirse el tiempo que están activas las tarjetas, reduciendo del mismo modo la batería que gastan. Además, la memoria disponible en las tarjetas es muy reducida, mientras que el ordenador desde el que se realice la monitorización tendrá una mayor capacidad.

Siguiendo el esquema propuesto, se diseña el código de las tarjetas de forma que conozcan unos ciertos códigos de operación, y a partir de ellos envíen los datos que reciben de una forma u otra. Para ello se dispondrá de los distintos tipos de paquete mostrados en la Tabla 8.

Operación	Campo 1	Campo 2	Campo 3	Campo 4	Campo 5	Campo 6	Campos 7...n
Escritura y lectura de I2C	Dirección del nodo	Tipo de operación	Repeated Start	Dirección del sensor	Número de datos a escribir	Número de datos a leer	Datos
Escritura de I2C	Dirección del nodo	Tipo de operación	Repeated Start	Dirección del sensor	Número de datos a escribir	Datos	
Lectura de I2C	Dirección del nodo	Tipo de operación	Dirección del sensor				
Lectura de ADC	Dirección del nodo	Tipo de operación	Canal del ADC				

Tabla 8. Tipos de paquete para comunicaciones inalámbricas

A continuación se detallan los campos que contienen los diferentes paquetes, todos ellos tienen tamaño de 1 byte, a excepción de los datos, que pueden ser mayores:

- Dirección del nodo: Dirección de la tarjeta específica a la que se dirige el mensaje.
- Tipo de operación: toma valores de 0x00 a 0x03 y permite a las tarjetas conocer el tipo de operación que se les solicita realizar
- *Repeated Start*: bandera que indica si debe utilizarse la opción de *repeated start* o no en las comunicaciones I2C
- Dirección del sensor: para las operaciones que tienen que ver con I2C, permite saber a qué dirección debe dirigirse la operación
- Número de datos a escribir: valor entero que muestra el tamaño de los datos del paquete
- Número de datos a leer: valor entero que proporciona información sobre el número de bytes que se deben recibir en la comunicación I2C
- Datos: para las comunicaciones I2C, bytes a enviar al sensor
- Canal del ADC: canal a leer del ADC

Cuando una de las tarjetas reciba un determinado paquete, comprobará en primer lugar si el paquete está dirigido a ella, en caso afirmativo, comprobará que operación debe realizar, y la llevará a cabo utilizando el resto de datos que contiene el paquete.

La capa de comunicaciones inalámbricas proporciona las funciones necesarias para crear los paquetes, dependiendo del tipo de operación que se desee realizar, así como para desempaquetar los paquetes recibidos de las tarjetas. Los paquetes creados serán pasados a la capa inferior, mientras que los recibidos se le proporcionarán a la superior.

5.2.3. Capa de gestión de tarjetas

Dicha capa será la que contenga la diferente información relativa a los nodos y las tarjetas conectadas a cada uno de ellos, así como los sensores presentes en las tarjetas. Contendrá información como la dirección de los sensores, la dirección de la tarjeta de sensores, que comandos hay que enviar para configurarlos o para leer sus datos, etc.

Para ello, será necesario definir una clase para cada sensor, así como otra clase para cada tarjeta de sensores. En ellas se encontrarán los datos importantes para trabajar con dichos componentes. Además, también se ha creado una clase “Nodo” que estará formada por las instanciaciones de las tarjetas de sensores correspondientes.

5.2.4. Capa de adaptación

Esta capa nos permitirá realizar las operaciones necesarias para pedir las distintas medidas a los nodos de la red de sensores, así como proporcionar estos datos a la interfaz gráfica, de forma que puedan ser mostrados por pantalla, y puedan crearse las notificaciones oportunas en caso de ser necesario.

La clase que la forma, Domotiksen, se encargará de realizar la configuración de los sensores al iniciar el programa, así como de realizar las peticiones de lectura de forma continua, con un breve lapso de tiempo entre repeticiones, de forma que se ahorre energía en las tarjetas. Una vez reciba los mensajes de respuesta a las peticiones, los datos recibidos se harán llegar a la interfaz gráfica, donde podrán ser mostrados.

5.2.5. Capa de usuario

Está formada por dos paquetes distintos, por una parte la interfaz gráfica, y por otra el sistema de notificaciones.

El sistema de notificaciones está formado por una sola clase, que se encarga de enviar un correo electrónico en caso de que alguno de los valores medidos no sea el esperado. Para ello se ha utilizado la librería Javamail

Por otra parte, la interfaz gráfica está compuesta por una ventana que contiene tres gráficas distintas, primero encontramos las medidas de temperatura, después las de humedad, y por último la lectura del sensor térmico. Para la representación de las gráficas, se ha empleado la librería JFreeChart.

6. Pruebas y resultados

De cara a comprobar que el diseño del sistema se ha realizado de forma correcta ha sido necesario en primer lugar fabricar un prototipo de las tarjetas desarrolladas (que pueden verse conectadas formando un nodo completo en la Figura 37) y sobre éste se han llevado a cabo diferentes pruebas para verificar su funcionamiento. En primer lugar, se han realizado pruebas para comprobar que las comunicaciones dentro del propio nodo inalámbrico eran correctas (comunicación con los diferentes sensores, expansores...).

Además se ha comprobado que las comunicaciones inalámbricas entre distintos nodos también podían llevarse a cabo sin problemas. Por último se ha comprobado la funcionalidad del conjunto completo, incluyendo nodo inalámbrico, nodo central y PC ejecutando el programa java.

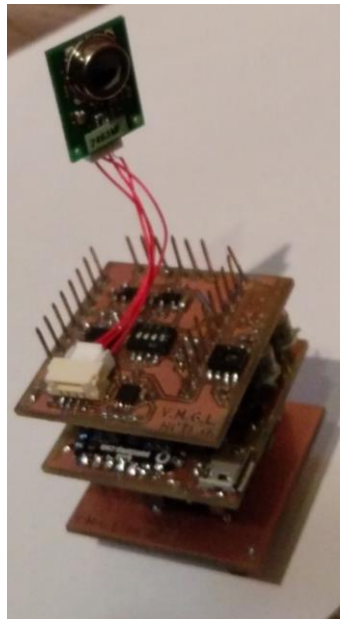


Figura 37. Nodo fabricado

6.1. Comunicaciones dentro del nodo inalámbrico

Para comprobar el correcto funcionamiento del nodo, se utiliza el software mostrado en el capítulo anterior. Para cada sensor, se utiliza un programa *Arduino* independiente, y se obtienen sus medidas.

6.1.1. Sensor de temperatura y humedad

Se carga en el microcontrolador el programa de control, y se comprueba mediante el uso del puerto serie qué valores lee el sensor.

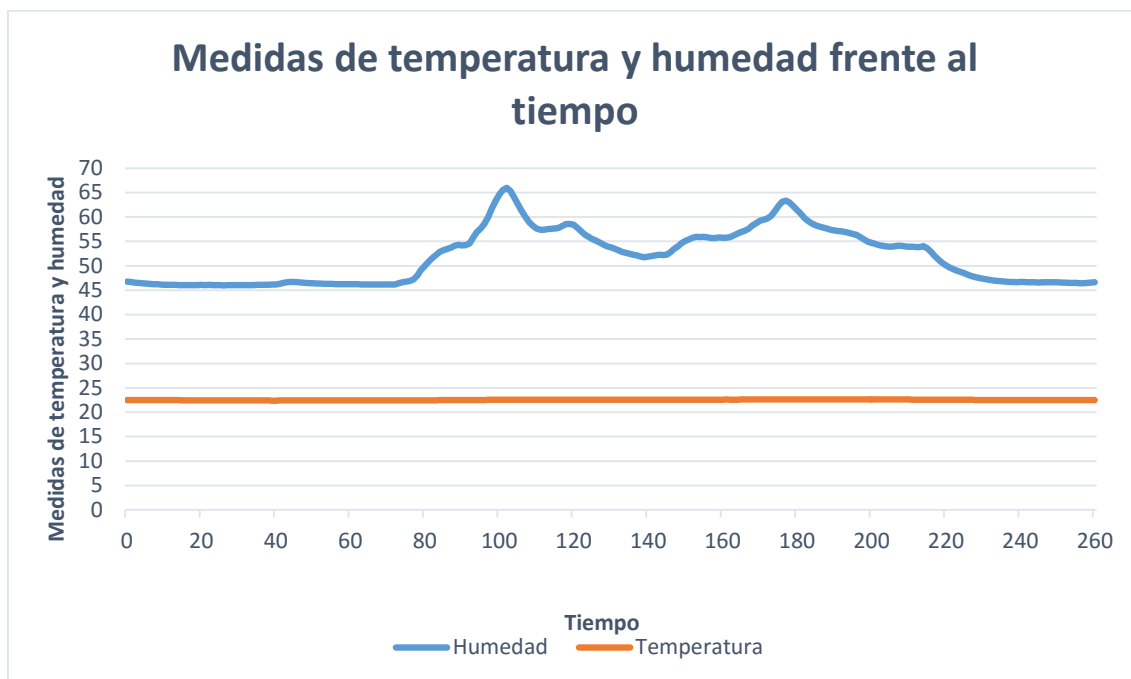


Figura 38. Medidas de temperatura y humedad frente al tiempo

La prueba se llevó a cabo dentro de una habitación y consistió en tomar varias mediciones con condiciones normales. Tras esto, se situó un vaso con agua debajo del sensor, y por último se volvió a retirar el vaso, dejando el sensor bajo condiciones normales. Los valores obtenidos fueron los mostrados en la Figura 38. En ella, puede observarse cuál fue el valor de temperatura obtenido (casi constante), que fue de entre 22,4 y 22,6 °C. Al mismo tiempo que se realizaba la prueba, un termómetro comercial estaba situado junto al nodo. En él se podía leer una temperatura de aproximadamente 21,9 °C, por tanto puede concluirse que la temperatura obtenida por el sensor nos permite tener una idea bastante acertada sobre la temperatura ambiente.

En el caso de la humedad, las oscilaciones son bastante amplias, aproximadamente varían entre 53 y 65 % estando el sensor situado junto al vaso de agua, es decir, se da una variación de ± 6 %, algo superior a la que aparece en la hoja de datos del sensor. Aun así, este nos permitiría saber si se produce algún cambio significativo en el ambiente (como si se abre una ventana, por ejemplo), ya que como puede observarse, bajo condiciones normales la medida que proporciona es estable.

6.1.2. Sensor térmico Omron D6T

Como ya se ha explicado en el capítulo 4, el sensor térmico es capaz de captar la temperatura de la superficie situada frente a él. De esta forma, si situamos delante del sensor dos objetos con diferente temperatura, a una distancia lo relativamente corta de forma que cubran todo el rango de visión del sensor, debería obtenerse la temperatura a la que se encuentran los objetos. En este caso, la prueba se ha realizado enfocando el sensor a una superficie cualquiera de la habitación, situando la mano delante del sensor, de forma que se cubra su campo de visión después, para terminar destapándolo de nuevo.

Los resultados obtenidos fueron los mostrados en la *Tabla 9*, donde cada fila representa uno de los píxeles del sensor, la primera columna representa la primera medida sobre la superficie, la segunda columna muestra los valores obtenidos tras cubrir el sensor, y la tercera

representa los valores obtenidos de nuevo al apartar la mano del sensor. Mientras se realizaban las pruebas, la medida de temperatura ambiente proporcionada por un termómetro comercial era de 22,5 °C. No se realizaron medidas sobre la temperatura de la piel con un sensor de temperatura comercial ya que simplemente se pretende comprobar que pueden detectarse cambios de temperatura considerables, que se producirán cuando una persona se encuentre frente al sensor, en comparación con la temperatura ambiente de una casa:

1ª medida sobre superficie	Medida cubriendo el sensor	2ª medida sobre superficie
20	33	20
20	33	20
21	33	21
23	33	23
20	33	20
21	33	21
24	34	24
24	34	24
20	33	20
24	33	22
26	33	26
26	34	26
21	33	20
25	33	22
27	33	26
27	33	26

Tabla 9. Prueba sobre el sensor Omron D6T

6.1.3. PCA9557

Para comprobar el correcto funcionamiento del expansor, se miden los valores que devuelve este y se muestran, como en las dos pruebas anteriores, utilizando el puerto serie, permitiendo comprobar así si los valores obtenidos son los esperados.

Para la tarjeta sobre la que se realizaban las pruebas, se eligió la dirección 0xF7. Dirigiendo las comunicaciones I2C a dicha dirección, y conociendo el valor programado en los interruptores presentes en la tarjeta, puedo comprobarse que los valores devueltos eran los esperados.

6.2. Comunicaciones inalámbricas

Para la realización de las pruebas sobre la comunicación inalámbrica de los nodos, se sitúan dos de ellos trabajando al mismo tiempo, uno actuando como emisor, y otro como receptor.

El primero enviará paquetes a una tasa fija de un paquete cada 100 ms, y el receptor los recibirá y mantendrá la cuenta del número de paquetes recibidos, de forma que pueda comprobarse si el sistema funciona correctamente o no.

Tras mantener la prueba sobre las comunicaciones inalámbricas durante 30 minutos (18000 paquetes), se calculan unas pérdidas del 7% de los paquetes. Por tanto, las pérdidas de paquetes no son despreciables, y se prevé que será necesario implementar un protocolo de reenvío de paquetes en futuras versiones del sistema para eliminar este problema.

6.3. Sistema completo

Mediante las pruebas realizadas sobre el sistema completo se pretende comprobar que funcionan por una parte, las comunicaciones inalámbricas y por otra el software de monitorización desarrollado, que si bien es más simple que el que se tendría que utilizar a la hora de comercializar el producto, servirá de base para éste.

Para realizar las pruebas sobre el sistema completo, se conecta uno de los nodos a un ordenador, de forma que actúe como módem, recibiendo los mensajes del ordenador y transmitiéndoselos al resto de nodos, y recibiendo las respuestas de éstos y enviándoselos al ordenador para que pueda representarlos.

Además, se sitúan unos umbrales de notificación bajos en el programa software, de forma que salten las alarmas del programa (el panel correspondiente cambia de color gris a amarillo, y se envía un correo electrónico a una dirección predefinida) bajo condiciones cercanas a las normales, permitiendo así comprobar la correcta funcionalidad del sistema.

Se han realizado tres pruebas, la primera de ellas, comprueba el funcionamiento del sensor de temperatura. Tras mantener el sistema bajo condiciones normales, se sitúa una fuente de calor cerca del nodo, y se comprueba si la medida de temperatura aumenta como era de esperar, y por otra parte si la notificación correspondiente se activa al sobrepasar el umbral. Los resultados pueden observarse en la Figura 39:

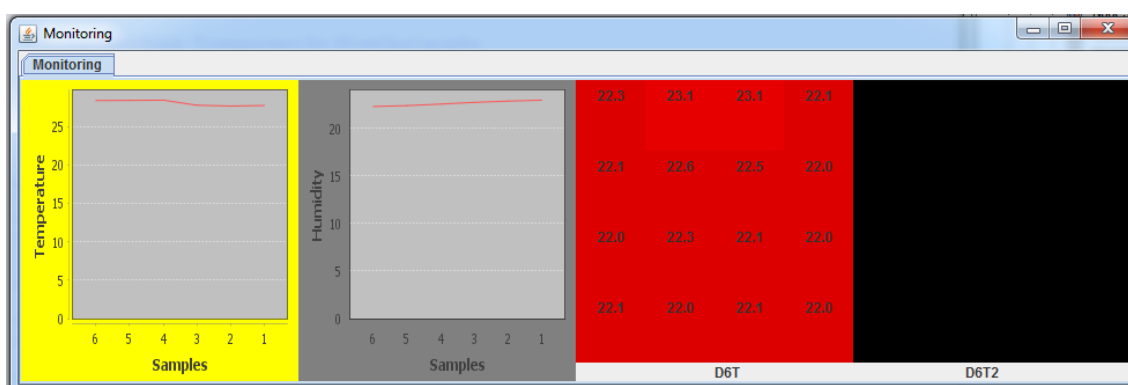


Figura 39. Prueba del sistema completo: temperatura

La segunda prueba se realiza de forma similar a la primera, con la diferencia de que, en lugar de situar una fuente de calor cerca del sensor, se sitúa una fuente de humedad. Los resultados se muestran en la Figura 40:

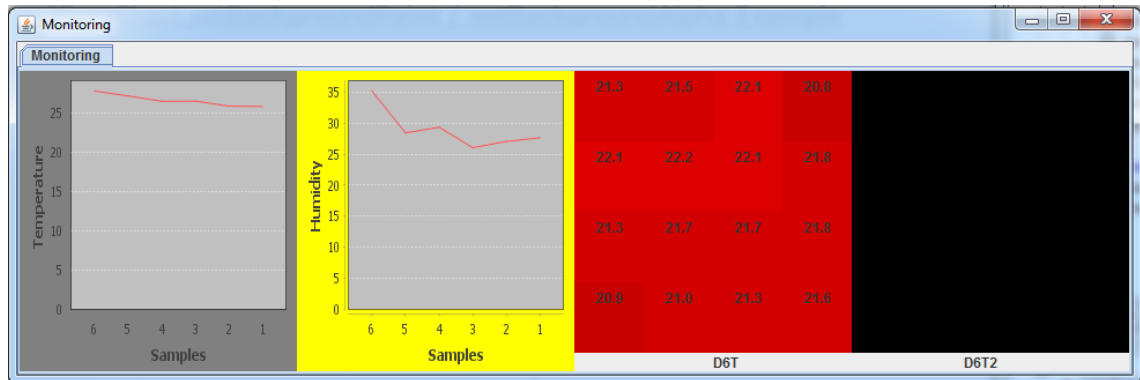


Figura 40. Prueba del sistema completo: humedad

Por último, se comprueba el funcionamiento del sensor de Omron. Para ello, se sitúa una persona delante del sensor, a una distancia de unos dos metros, simulando lo que podría ser una persona tumbada en una cama. Se realizaron dos medidas distintas, cambiando la posición de la persona en unos 30 cm entre ambas medidas. Los resultados obtenidos se muestran en la Figura 41 y la Figura 42:

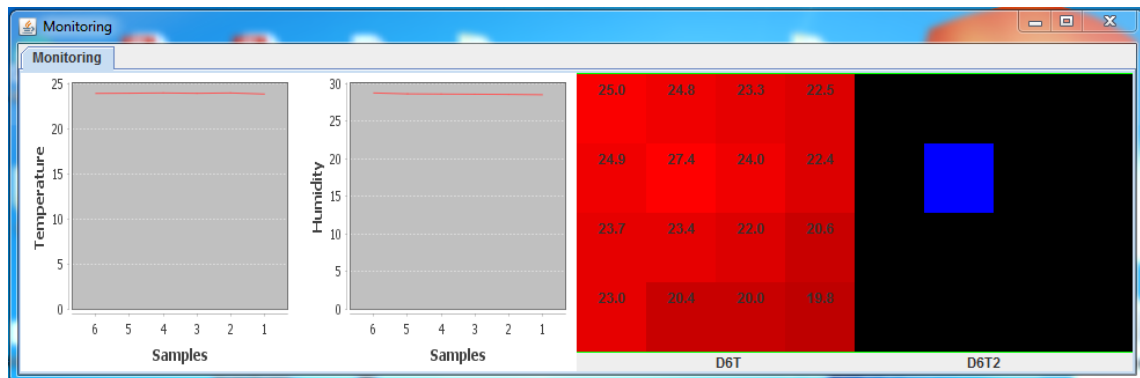


Figura 41. Prueba del sistema completo: Omron (primera medida)

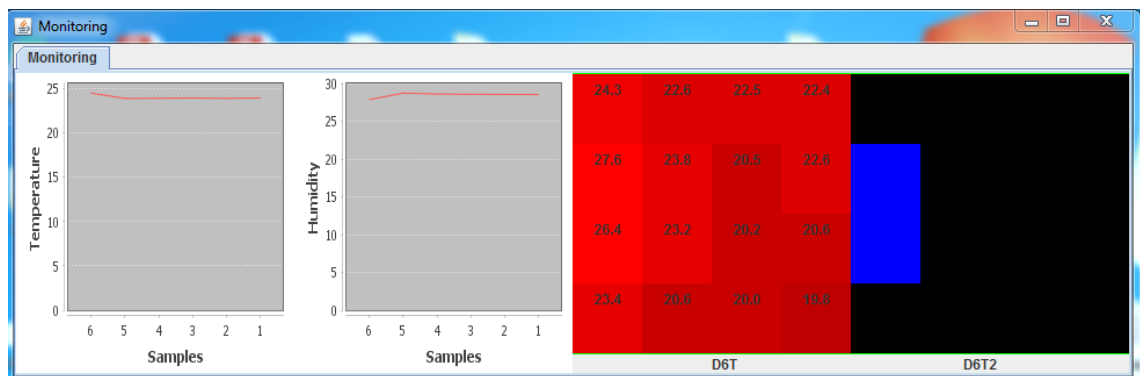


Figura 42. Prueba del sistema completo: Omron (segunda medida)

7. Conclusiones

En la actualidad, cada día hay más personas mayores que tienen que vivir solas en sus hogares. El aumento en esta cantidad de personas, junto al envejecimiento de la población, hace que sea necesario buscar alternativas para su cuidado. Una de estas alternativas es el uso de una red de sensores que permita monitorizar el hogar de la persona, comprobando así que no hay ningún problema.

En el presente TFM se han realizado los diseños necesarios para la creación de una red de sensores inalámbrica orientada a la monitorización de hogares para personas dependientes. Por ello, se ha tenido especialmente en cuenta a la hora de desarrollar el sistema los consumos de los componentes, y su tamaño.

Además del diseño hardware, también se ha desarrollado un programa software que permitiese comprobar el correcto funcionamiento del sistema, así como servir de base para futuros desarrollos.

Por último, se han realizado pruebas tanto sobre las tarjetas diseñadas por separado, como sobre el sistema completo, permitiendo de este modo asegurar que el diseño se ha realizado de forma correcta.

Como se ha mostrado en el capítulo 6, mediante las pruebas realizadas ha podido comprobarse que las distintas partes del diseño funcionan correctamente y que, por tanto, se ha completado el proyecto de forma satisfactoria.

8. Trabajo Futuro

Como trabajo futuro de este proyecto se propone:

- Fabricación de nuevas tarjetas que sigan el *pinout* elegido finalmente en el apartado 4.2.
- Diseño de nuevas tarjetas que puedan incorporar sensores diferentes, como por ejemplo sensores de gas, o de vibración.
- Diseño de un programa de monitorización más sofisticado, que permita variar desde su interfaz diferentes parámetros que pudieran ser útiles (valores de *trigger* para las notificaciones, dirección de email con la que contactar, inicio de sesión en el programa...).
- Encriptación de los mensajes compartidos de forma inalámbrica.
- Análisis de consumo.
- Puesto que el producto final que intenta desarrollarse debe ser comercial, durante el proceso de diseño se ha tratado de elegir distintos componentes que permitieran cumplir los estándares necesarios para ello (que se encuentran brevemente descritos en el Anexo I). Será necesario comprobar que finalmente se cumplen.

Anexo I: Certificaciones

Cómo se ha comentado con anterioridad, el objetivo del presente TFM es desarrollar un prototipo para un producto que finalmente tendrá fines comerciales. Por tanto, es importante que dicho producto cumpla las normativas vigentes de cara a la comercialización en los lugares objetivo. En este caso, se desea vender el producto en Europa y en los Estados Unidos, por tanto, debido al tipo de producto del que se trata, deberá cumplir con los estándares CE en Europa y FCC en Estados Unidos.

Más específicamente, para ser comercializado en Europa deberá cumplir con el estándar RohS, así como el R&TTE.

Las directivas a cumplir en Europa son por tanto:

- R&TTE: DIRECTIVA 1999/5/CE.
- RohS: DIRECTIVA 2011/65/UE.

En general, mediante el cumplimiento de la FCC y la R&TTE se asegura que el dispositivo desarrollado no va a causar interferencia a otros dispositivos cercanos (muy importante, ya que está pensado para utilizarse en hogares, de forma que no puede interferir con otras señales como las de radio o de televisión), y que será capaz de aceptar las interferencias que otros puedan causarle.

Para ello, la FCC proporciona ciertos datos sobre los valores que deben obtenerse en cuanto a potencia de campo radiado, así sobre cómo deben llevarse a cabo estas pruebas. La CE proporciona a su vez una serie de estándares armonizados que deberían seguirse para comprobar si nuestro dispositivo cumple con las normas de la directiva o no. En cualquier caso, tanto una directiva como la otra permiten que el fabricante realice las pruebas que el mismo decida para comprobar si se cumplen o no las normas, siempre y cuando se demuestre que las pruebas son válidas y se justifique su uso.

Anexo II: Glosario

TFM – Trabajo Fin de Máster
WSN – Wireless Sensor Network
COTS – Commercial Off-The-Shelf
JTAG – Joint Test Action Group
SPI – Serial Peripheral Interface
I2C – Inter-Integrated Circuit
MAC – Media Control Access
OS – Operating System
GPS – Global Positioning System
CRC – Cyclic Redundancy Check
WLAN – Wireless Local Area Network
TCP – Transmission Control Protocol
IP – Internet Protocol
USB – Universal Serial Bus
UART – Universal Asynchronous Receiver-Transmitter
PWM – Pulse-Width Modulation
ADC – Analog to Digital Converter
SBW – Spy-Bi-Wire
BSL – Boot Strap Loader
SWAP – Simple Wireless Abstract Protocol
XML – Extensible Markup Language

X. Referencias

1. Naciones Unidas, Departamento de Asuntos Económicos y sociales (2007). **El desarrollo en un mundo que envejece**. *Estudio Económico y Social Mundial 2007*.
2. Jiun-Ren Ding. **Bed Status Detection for Elder-care Center**. *Department of Digital Home Systems Technology, Home Networking Technology Center, Industrial Technology Research Institute, Tainan, Taiwan (R.O.C.)*.
3. Sandra Nava-Muñoz, Alberto L. Morán y Jesús Favela (2010). **Providing Awareness of Elder's Situations of Care through a Context-Aware Notification Enviroment**. *Sixth International Conference on Intelligent Enviroments*.
4. Isabel Marcelino y, António Pereira (2009). **Elder Care Modular Solution**. *Second International Conference on Advances in Human-Oriented and Personalized Mechanism, Technologies and Services*.
5. Gaozhi Xu, Jianqing Li, Lei Wang, Jianfeng Wu, Jianfeng Wu, Kan Luo, Yinggao Yue y Feng Ran (2013). **Hardware Design of A Body Sensor Network System Used for Elder Care**. *Ninth International Conference on Natural Computation (ICNC)*.
6. Cheonshu Park y Jaehong Kim (2011). **A Location and Emergency Monitoring System for Elder Care using ZigBee**. *Seventh International Conference on Mobile Ad-hoc and Sensor Networks*.
7. Cheonsu Park, Sangseung Kang, Jaehong Kim and Jinhwan Oh (2012). **A Study on Service Robot System for Elder Care**. *Ninth International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*.
8. Isabel Marcelino, João Barroso, José Bulas Cruz y António Pereira (2008). **Elder Care Architecture**. *The Third International Conference on Systems and Networks Communications*.
9. Farhan I. Simjee y Pai H. Chou (2008). **Efficient Charging of Supercapacitors for Extended Lifetime of Wireless Sensor Nodes**. *IEEE TRANSACTIONS ON POWER ELECTRONICS, VOL. 23, NO. 3*.
10. Kevin M. Farinholt, Gyuhae Park, y Charles R. Farrar (Julio 2009). **RF Energy Transmission for a Low-Power Wireless Impedance Sensor Node**. *IEEE SENSORS JOURNAL, VOL. 9, NO. 7*.
11. Hannes Reinisch, Stefan Gruber, Hartwig Unterassinger, Martin Wiessflecker, Günter Hofer, Wolfgang Pribyl y Gerald Holweg (Julio 2011). **An Electro-Magnetic Energy Harvesting SystemWith 190 nW Idle Mode Power Consumption for a BAW Based Wireless Sensor Node**. *IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 46, NO. 7*.
12. Adriano B. da Cunha, Breno R. de Almeida y Diógenes C. da Silva (Junio 2009). **Remaining Capacity Measurement and Analysis of Alkaline Batteries for Wireless Sensor Nodes**. *IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, VOL. 58, NO. 6*.
13. Cedric Walravens y Wim Dehaene (Febrero 2014). **Low-Power Digital Signal Processor Architecture for Wireless Sensor Nodes**. *IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 22, NO. 2*.

14. Maria T. Penella y Manel Gasulla (Abril 2010). **Runtime Extension of Low-Power Wireless Sensor Nodes Using Hybrid-Storage Units**. *IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT*, VOL. 59, NO. 4.
15. Y.K. Tan, K.Y. Hoe y S.K. Panda (2006). **Energy Harvesting using Piezoelectric Igniter for Self-Powered Radio Frequency (1F) Wireless Sensors**.
16. Marco Tacca, Paolo Monti y Andrea Fumagalli (Julio 2007). **Cooperative and Reliable ARQ Protocols for Energy Harvesting Wireless Sensor Nodes**. *IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS*, VOL. 6, NO. 7.
17. **Wireless Sensor Networks**. *Lecture 8. CS252/Hill*.
18. Jonathan R. Agre, Loren P. Clare, Gregory J. Pottie y Nikolai P. Romanov. **Development Platform for Self-Organizing Wireless Sensor Networks**.
19. Jason L. Hill y David E. Culler (2002). **MICA: A WIRELESS PLATFORM FOR DEEPLY EMBEDDED NETWORKS**. *Universidad de California, Berkeley*.
20. **Mica2 wireless sensor network datasheet**. *www.xbow.com (Part number 6020-0042-04)*.
21. Hector Abrach, Jim Carlson, Hui Dai, Jeff Rose, Anmol Sheth, Brian Shucker y Richard Han (Abril 2003). **MANTIS: System Support for Multimodal Networks of In-situ Sensors**. *University of Colorado, Department of Computer Science*.
22. Martin Ling y Steven Wong. **ProSpeckz III – Design and Implementation**. *University of Edinburgh*.
23. Victor Shnayder, Bor-rong Chen, Konrad Lorincz, Thaddeus R. F. Fulford-Jones y Matt Welsh (2005). **Sensor Networks for Medical Care**. *Technical Report TR-08-05, Division of Engineering and Applied Sciences, Harvard University*.
24. Andre Barroso, Jonathan Benson, Tina Murphy, Utz Roedig, Cormac Sreenan, John Barton, Stephen Bellis, Brendan O’Flynn and Kieran Delaney (Noviembre 2004). **Demo Abstract: The DSYS25 Sensor Platform**. *SenSys’04, Baltimore, Maryland, USA*.
25. Carlos Hernández Gañán (Marzo 2008). **Energy Management Architecture for Wireless Sensor Networks**. *Koninklijke Philips Electronics*.
26. Michale Baar, Enrico Köppe, Achim Liers y Jochen Schiller (Marzo 2007). **Poster Abstract: The ScatterWeb MSB-430 Platform for Wireless Sensor Networks**. *SICS Contiki Workshop*.
27. Adrian Burns, Barry R. Greene, Michael J. McGrath, Terramce J. O’Shea, Benjamin Kuris, Steven M. Ayer, Florin Stroiescu y Victor Cionca (Septiembre 2010). **SHIMMER™ – A Wireless Sensor Platform for Noninvasive Biomedical Research**. *IEEE SENSORS JOURNAL*, VOL. 10, NO. 9.
28. **Neomote Wireless Sensor Network datasheet**. *www.metronomesystems.com. Metronome Systems*.
29. **Arduino BT datasheet**. *www.arduino.cc*.
30. **Indriya_DP_03A14 datasheet** (Marzo 2011). *www.indrion.co.uk. Indrion*.

31. Maxim Buevich, Niranjini Rajagopal y Anthony Rowe. **Hardware Assisted Clock Synchronization for Real-Time Sensor Networks**. *Electrical and Computer Engineering Department. Carnegie Mellon University. Pittsburgh, USA*.
32. **Wismote Wireless Sensor Networks datasheet**. *www.aragosystems.com. Arago Systems*.
33. **WiSense Wireless Sensor Node datasheet** (2004). *www.winsense.in. WiSense*.
34. **PanStamp NRG wiki page**. *www.panStamp.com. PanStamp*.
35. Gourab Sen Gupta y Mark Hetherington (2012). **Miniaturisation of Wireless Sensor Nodes for Smart Digital Home**.
36. N.K. Suryadevara, A. Gaddam, R.K. Rayudu y S.C. Mukhopadhyay (Marzo 2012). **Sensors and Actuators A: Physical**. *www.elsevier.com. Elsevier*.
37. Anuroop Gaddam, Subhas Chandra Mukhopadhyay y Gourab Sen Gupta (Marzo 2011). **Elder Care Based on Cognitive Sensor Network**. *IEEE SENSORS JOURNAL, VOL. 11, NO. 3*.
38. Francisco J. Fernández-Luque, Juan Zapata y Ramón Ruiz (Septiembre 2010). **A System for Ubiquitous Fall Monitoring at Home via a Wireless Sensor Network**. *32nd Annual international Conference of the IEEE EMBS. Buenos Aires, Argentina*.
39. G. Virone, A. Wood, L. Selavo, Q. Cao, L. Fang, T. Doan, Z. He, R. Stoleru, S. Lin y J.A. Stamkovic. **An Advanced Wireless Sensor Network for Health Monitoring**.
40. Paul Cuddihy, Jenny Weisenberg, Catherine Graichen y Meena Ganesh (Junio 2007). **Algorithm to Automatically Detect Abnormally Long Periods of Inactivity in a Home**. *HealthNet'07. San Juan, Puerto Rico, USA*.
41. Rasika S. Ransing y Manita Rajput (2015). **Smart Home for Eldery Care, based on Wireless Sensor Network**. *International Conference on Nascent Technologies in the Engineering Field*.